OPTIMAL DESIGN OF OBSERVER/CONTROLLERS

Robert Ernest LaRock

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

OPTIMAL DESIGN OF OBSERVER/CONTROLLERS

by

Robert Ernest LaRock

December 1974

Thesis Advisor: D. E. Kirk

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER

2. GOVT ACCESSION NO.

3. RECIPIENT'S CATALOG NUMBER

4. TITLE (and Subtitle)

Optimal Design of Observer/Controllers

5. TYPE OF REPORT & PERIOD COVERED

Electrical Engineer; December 1974

6. PERFORMING ORG. REPORT NUMBER

7. AUTHOR(s)

Robert Ernest LaRock

8. CONTRACT OR GRANT NUMBER(s)

9. PERFORMING ORGANIZATION NAME AND ADDRESS

Naval Postgraduate School
Monterey, California 93940

10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS

11. CONTROLLING OFFICE NAME AND ADDRESS

Naval Postgraduate School
Monterey, California 93940

12. REPORT DATE

December 1974

13. NUMBER OF PAGES

111

14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)

Naval Postgraduate School
Monterey, California 93940

15. SECURITY CLASS. (of this report)

Unclassified

15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Optimal Design
Observer
Controller
Quadratic Performance Measure

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A new technique for full-order and reduced-order observer design which is based on the minimization of an integral square-error performance measure is developed for the integral of time-multiplied-square-error performance measure. The technique is extended to include the design of full-order and reduced-order combined observer/controllers. The technique is well suited to the design of higher order, multiple-

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE
(Page 1) S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block #20 continued

input, multiple-output systems where traditional methods become difficult to apply.

Optimal Design
of
Observer/Controllers

by

Robert Ernest LaRock
Lieutenant, United States Navy
B. S., Purdue University, 1969

Submitted in partial fulfillment of the
requirements for the degree of

ELECTRICAL ENGINEER

from the

NAVAL POSTGRADUATE SCHOOL
December 1974

ABSTRACT

A new technique for full-order and reduced-order observer design which is based on the minimization of an integral square-error performance measure is developed for the integral of time-multiplied-square-error performance measure. The technique is extended to include the design of full-order and reduced-order combined observer/controllers. The technique is well suited to the design of higher order, multiple-input, multiple-output systems where traditional methods become difficult to apply.

TABLE OF CONTENTS

ACKNOWLEDGEMENT

The author wishes to gratefully acknowledge the guidance of Professor D. E. Kirk in the preparation of this thesis. The author gives his deepest appreciation to his wife and children, without whose patient understanding the program could not have been completed.

I. INTRODUCTION

In feedback control systems, the plant control is often a function of the current plant state vector. A knowledge of the state vector is therefore required for the controller. In most systems, however, the entire state vector is not available through direct measurement. It is therefore necessary to estimate the unknown states. An observer is a device which can be used to reconstruct all or part of the state vector [1].

Traditionally, observers and controllers have been designed separately. Time and frequency domain specifications are satisfied through the selection of controller feedback gains as if all of the plant states are available. The observer is designed to estimate the plant states by proper selection of the observer gain matrix. The observer output is used to supply estimates of the missing states to the controller [2].

In this thesis, after presenting the necessary background material, a method for designing observers which minimizes a quadratic performance measure is presented. The method is also extended to the simultaneous design of the observer and the controller. For simplicity, only linear, time-invariant systems are considered. The design method relies on the use of a digital computer algorithm, however, the computations are performed off-line. Numerical examples are presented in order to illustrate the new technique.

II. MATHEMATICAL BACKGROUND

A. TRADITIONAL DESIGN TECHNIQUES

The dynamic characteristics of an n^{th}-order, linear, time-invariant, free system can be described by the linear, first order differential equations

$$\dot{\underline{s}}(t) = \underline{Hs}(t) \tag{1}$$

where $\underline{s}(t)$ is an nx1 vector and $\underline{H}$ is a constant nxn matrix. The response of the system to any non-zero initial condition is determined by the elements of the matrix $\underline{H}$ which are, in turn, a function of the system configuration and physical constants. By adjusting system parameters, a suitable $\underline{H}$ matrix is realized which causes the system to meet certain system performance specifications.

Traditionally, performance is specified as a consistent set of time and frequency domain criteria. Typical specifications are given in terms of rise time, settling time, percent overshoot, steady-state error, gain margin, phase margin, and bandwidth. These criteria are then translated into a control strategy through the use of some relatively simple mathematical tools; generally by trial-and-error techniques. Appropriate assumptions are usually made to attempt to approximate complex, higher-order systems by simpler, lower-order models that can be handled with greater ease.

This approach, although empirical in nature, and relying heavily on the ingenuity and system know-how of the

designer, is relatively straightforward for simple, single-input, single-output systems. Many modern systems are highly complex, however, with multiple inputs and multiple outputs. These systems do not readily lend themselves to the mathematical techniques alluded to earlier and often performance is not easily specified in terms of the traditional criteria. It is therefore desirable to have an alternative method for prescribing desired system performance.

B. THE QUADRATIC PERFORMANCE MEASURE

One such index of system performance is the quadratic performance measure

$$J = \int_0^\infty t^r \underline{s}^T(t) \underline{Q} \underline{s}(t) dt \tag{2}$$

which assigns a positive scalar value to the infinite time integral of the time-multiplied quadratic form in $\underline{s}(t)$ and $\underline{Q}$. The elements of the nxn matrix $\underline{Q}$ are specified to determine the relative contributions to J of the elements of the vector $\underline{s}(t)$. The value of the time-multiplier exponent, r, establishes the relative contributions to J of the earlier and later values of the elements of $\underline{s}(t)$. Interpreting J as a cost associated with a particular system configuration, it becomes the task of the designer to find values for the elements of $\underline{H}$ which are at his disposal that yield the lowest value for J. This choice for $\underline{H}$ is said to be optimal since it minimizes the cost.

Examples of systems to which a performance measure of the form of (2) can be applied are the observer error state

equations

$$\dot{\underline{e}}(t) = \underline{Fe}(t) \tag{3}$$

and the state-error differential equations of a combined plant-observer/controller

$$\begin{bmatrix} \dot{\underline{x}}(t) \\ \hline \dot{\underline{e}}(t) \end{bmatrix} = \left[\begin{array}{c|c} \underline{A}+\underline{BK} & \underline{BK} \\ \hline \underline{0} & \underline{F} \end{array}\right] \begin{bmatrix} \underline{x}(t) \\ \hline \underline{e}(t) \end{bmatrix} . \tag{4}$$

Equations (3) and (4) will be developed in detail in Sections III and IV, respectively.

C. EVALUATING THE QUADRATIC PERFORMANCE MEASURE

In Ref. 3 it is shown that if (1) is a stable,[1] linear system and if $\underline{Q}$ is any positive definite matrix, the performance measure (2) can be expressed as

$$J = (-1)^{r+1} r! \underline{s}^T(0) \underline{V}_{r+1} \underline{s}(0) . \tag{5}$$

The real symmetric matrix $\underline{V}_{r+1}$ is the unique solution of the simultaneous linear, algebraic matrix equations

$$\begin{aligned} \underline{H}^T \underline{V}_{s+1} &+ \underline{V}_{s+1}\underline{H} = \underline{V}_s \qquad s = 1,2,\ldots r \\ &\vdots \\ \underline{H}^T \underline{V}_1 &+ \underline{V}_1 \underline{H} = -\underline{Q} . \end{aligned} \tag{6}$$

It can be shown [4] that if $\underline{H}$ is a stability matrix and if $\underline{Q}$ is positive definite then the matrix $\underline{V}_1$ is also positive definite. The argument can easily be extended to show that $(-1)^{r+1} r! \underline{V}_{r+1}$ is also positive definite.

---

[1]The system (1) is stable if $\underline{H}$ is a stability matrix; that is if, and only if, all eigenvalues of $\underline{H}$ have negative real parts.

The details of the derivation of (5) and (6) are given in Appendix A. The solution to (6) can be obtained by iteratively solving the r+1 sets of n(n+1)/2 simultaneous linear equations[2] or by a technique suggested in Ref. 3. The details of this technique are given in Appendix B.

Evaluation of the performance measure (2) using equations (5) and (6) eliminates the need for determining the explicit time response of the system but requires knowledge of the initial condition, $\underline{s}(0)$. Since *a priori* knowledge of this vector is not generally available, the question arises as to how to pick $\underline{s}(0)$ in order to arrive at a meaningful value for J.

One technique, suggested in Ref. 5, is to determine the average value of J over a linearly independent set of initial condition vectors. This is equivalent to assuming that the initial condition vector is uniformly distributed on the surface of a unit hypersphere and results in a design which is optimal in an average sense. Another technique, suggested in Ref. 6, is to determine the largest possible J for every $\underline{s}(0)$ within or on a hypersphere of radius ρ and select the configuration which yields the smallest of these maximum values. This results in a design which is optimal in a worst case sense. For the latter scheme, it is shown in Appendix C that for a given design parameter set {P}, the largest value for J is given by J_1 where

---

[2]Since the matrices $\underline{V}_1$, $\underline{V}_2,\ldots,\underline{V}_{r+1}$ are each symmetric, only n(n+1)/2 elements of each need to be solved for.

$$J_1(\{P\}) = \lambda_{max}[(-1)^{r+1} r! \underline{V}_{r+1}(\{P\})] \tag{7}$$

and where $\lambda_{max}[\cdot]$ represents the largest eigenvalue of $[\cdot]$. Here, the functional dependence of J_1 upon the design parameter set $\{P\}$ is stressed.

D. MINIMIZING THE QUADRATIC PERFORMANCE MEASURE

To this point we have assumed that the design parameter set $\{P\}$ is unbounded. In general, however, we must allow for the possibility of a constrained $\{P\}$. There may be explicit as well as implicit constraints. The explicit constraints are those which are the result of the physical limitations on the system components while the implicit constraints are those which result from restrictions placed on system performance by the designer

In either case, the constraining relationships must be included in the problem formulation. One means of accomplishing this is to add a large penalty to the cost of a given configuration if that configuration requires the violation of any of the constraints. With this in mind we define the optimal parameter set $\{P^*\}$ as that parameter set which minimizes $J_1(\{P\})$ and the minimum value of $J_1(\{P^*\})$ as J^*; that is

$$J^* \triangleq J_1(\{P^*\}) = \min_{\{P\}} J_1(\{P\}).$$

Therefore[3]

[3]Since r! is a constant in any given design specification, it does not contribute to the optimization of the parameter set and can therefore be deleted from the performance measure.

$$J^* = \min_{\{P\}}\{\lambda_{max}[(-1)^{r+1}\underline{V}_{r+1}(\{P\})]\}. \qquad (8)$$

This indicates that the following iterative procedure can be used to find $\{P^*\}$ once r and $\underline{Q}$ have been specified.

1. Select an initial estimate for {P} insuring that any constraints imposed are satisfied and that $\underline{H}$ is a stability matrix.

2. Using equation (6) or the method described in Appendix B, find

$$(-1)^{r+1}\underline{V}_{r+1}(\{P\}).$$

3. Evaluate

$$J_1(\{P\}) = \lambda_{max}[(-1)^{r+1}\underline{V}_{r+1}(\{P\})].$$

4. Through some suitable minimizing search routine[4] select a new set of parameters {P}. If the constraints are not satisfied, add a penalty to the cost and select a new parameter set.

5. Repeat steps two through four until some criterion for ending the search is satisfied.

---

[4]For the examples used in this thesis, the subroutine "DIRECT" (NPS Computer Facility) that performs the pattern search method of Hooke and Jeeves was used.

III. OBSERVER DESIGN

In the design of feedback control systems it is often desirable to have information about plant states which is not normally available through direct measurement. It is sometimes possible to construct a system called an observer which can estimate a linear transformation of the plant state and, if the transformation is invertible, it is then possible to reconstruct the plant state vector itself.

Traditionally, observer design has been accomplished through the use of techniques such as those described in Section IIA. In this section, techniques for the design of full-order and reduced-order observers by using the quadratic performance measure as the design specification are presented.

A. FULL-ORDER OBSERVER DESIGN

We consider the time-invariant, linear system

$$\begin{aligned} \dot{\underline{x}}(t) &= \underline{A}\underline{x}(t) + \underline{B}\underline{u}(t) \\ \underline{y}(t) &= \underline{C}\underline{x}(t) \end{aligned} \tag{9}$$

where $\underline{x}(t)$ is the nx1 state vector, $\underline{y}(t)$ is the mx1 output vector, $\underline{u}(t)$ is an ℓx1 control vector, $\underline{A}$ is an nxn constant matrix, $\underline{B}$ is an nxℓ constant matrix, and $\underline{C}$ is an mxn constant matrix. We consider also the full-order identity observer shown in Fig. 1. The observer state equations as developed in Appendix D are given by

$$\dot{\underline{z}}(t) = (\underline{A}-\underline{G}\underline{C})\underline{z}(t) + \underline{G}\underline{y}(t) + \underline{B}\underline{u}(t) \tag{10}$$

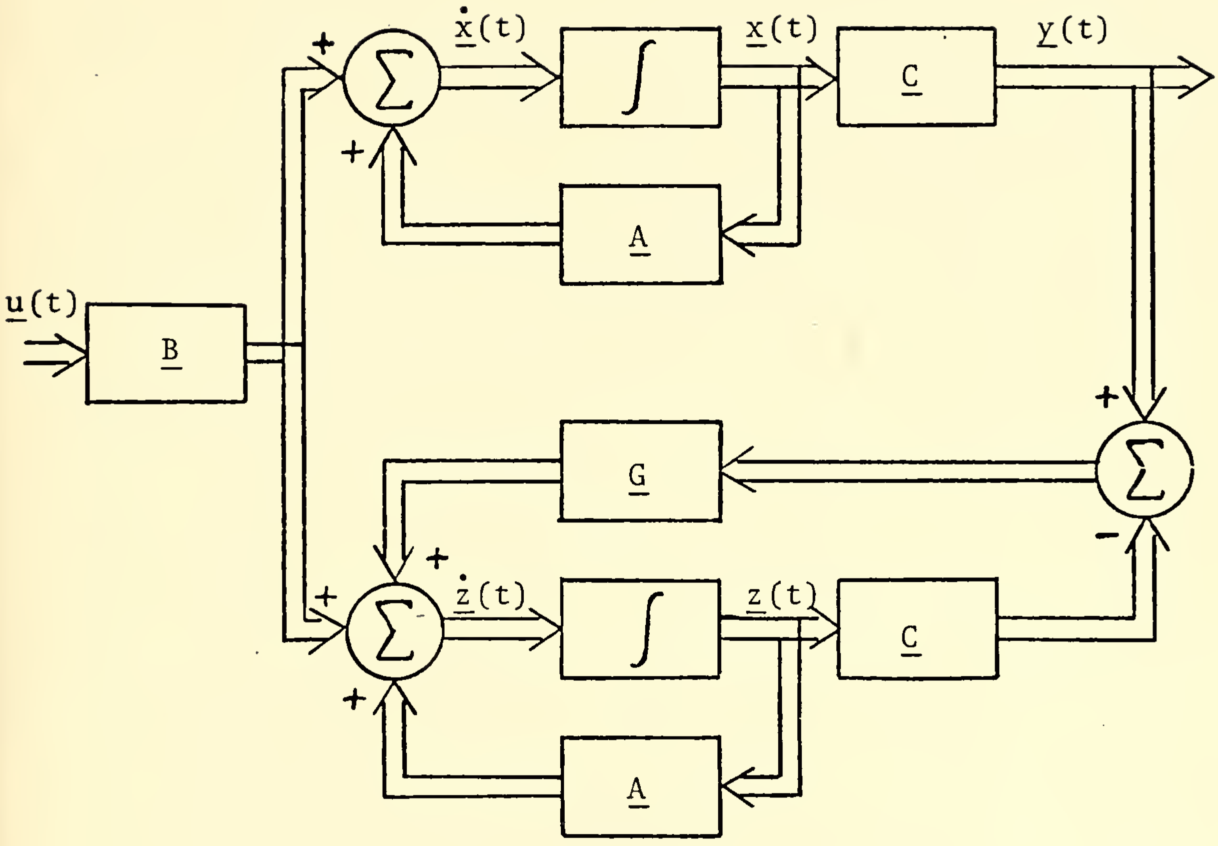


Figure 1. Full-Order Observer.

where $\underline{z}(t)$ is the nx1 observer state vector, $\underline{G}$ is the nxm observer gain matrix (in this case the elements of $\underline{G}$ constitute the design parameter set {P}), and where it is assumed that $(\underline{C},\underline{A})$ is observable.

Defining the error state as

$$\underline{e}(t) \triangleq \underline{z}(t) - \underline{x}(t)$$

and differentiating with respect to time yields

$$\dot{\underline{e}}(t) = \dot{\underline{z}}(t) - \dot{\underline{x}}(t).$$

Substituting (9) and (10) for $\dot{\underline{x}}(t)$ and $\dot{\underline{z}}(t)$, respectively, and solving for $\dot{\underline{e}}(t)$ yields

$$\dot{\underline{e}}(t) = (\underline{A}-\underline{GC})\underline{e}(t).$$

We define the matrix $\underline{F}$ as

$$\underline{F} \triangleq (\underline{A}-\underline{GC}) \tag{11}$$

which yields

$$\dot{\underline{e}}(t) = \underline{Fe}(t). \tag{12}$$

We define the quadratic performance measure as

$$J \triangleq \int_0^\infty t^r \underline{e}^T(t)\underline{Qe}(t)dt. \tag{13}$$

The system (12) and the performance measure (13) have the same form as (1) and (2), respectively. Equation (13) can therefore be rewritten as

$$J = (-1)^{r+1} r! \underline{e}^T(0)\underline{V}_{r+1}\underline{e}(0)$$

where, by using (6),

$$\underline{F}^T\underline{V}_{s+1} + \underline{V}_{s+1}\underline{F} = \underline{V}_s \qquad s = 1,2,\ldots,r$$

$$\vdots$$

$$\underline{F}^T\underline{V}_1 + \underline{V}_1\underline{F} = -\underline{Q}.$$

Furthermore, the algorithm described in Section IID can be applied to find $\underline{G}^*$, the observer gain matrix which is optimal with respect to the performance measure (13).

B. REDUCED-ORDER OBSERVER DESIGN

We again consider the time-invariant, linear system given by

$$\begin{aligned} \dot{\underline{x}}(t) &= \underline{A}\underline{x}(t) + \underline{B}\underline{u}(t) \\ \underline{y}(t) &= \underline{C}\underline{x}(t). \end{aligned} \tag{14}$$

We define the vector $\tilde{\underline{x}}(t)$ as

$$\tilde{\underline{x}}(t) \triangleq \begin{bmatrix} \underline{y}(t) \\ \hline \underline{w}(t) \end{bmatrix}$$

where $\underline{w}(t)$ is an (n-m)x1 vector given by

$$\underline{w}(t) = \underline{D}\underline{x}(t) \tag{15}$$

and where $\underline{D}$ is an (n-m)xn matrix, arbitrarily selected so that

$$\begin{bmatrix} \underline{C} \\ \hline \underline{D} \end{bmatrix}^{-1}$$

exists. By proper selection of the state variables $\underline{C}$ will have the form

$$\underline{C} = [\ \underline{I} \ \vdots \ \underline{0}\ ].$$

A notationally convenient choice for $\underline{D}$ is

$$\underline{D} = [\ \underline{0} \ \vdots \ \underline{I}\ ].$$

We can find $\hat{\underline{w}}(t)$, the estimate for $\underline{w}(t)$ by constructing the reduced-order observer as shown in Fig. 2. For the above choice for $\underline{C}$ and $\underline{D}$, the observer state equations developed in Appendix D can be written as

$$\dot{\underline{z}}(t) = (\underline{A}_{22}-\underline{G}\underline{A}_{12})\underline{z}(t) + [\underline{A}_{21}-\underline{G}\underline{A}_{11} + (\underline{A}_{22}-\underline{G}\underline{A}_{12})\underline{G}]\underline{y}(t)$$

$$+ (\underline{B}_2-\underline{G}\underline{B}_1)\underline{u}(t) \qquad (16)$$

where

$$\hat{\underline{w}}(t) = \underline{z}(t) + \underline{G}\underline{y}(t) \qquad (17)$$

and where the matrices $\underline{A}$ and $\underline{B}$ are partitioned as

$$\underline{A} = \left[\begin{array}{c|c} \underline{A}_{11} & \underline{A}_{12} \\ \hline \underline{A}_{21} & \underline{A}_{22} \end{array}\right], \quad \underline{B} = \left[\begin{array}{c} \underline{B}_1 \\ \hline \underline{B}_2 \end{array}\right] .$$

Here, $\underline{A}_{11}$ is an mxm matrix, $\underline{A}_{12}$ is an mx(n-m) matrix, $\underline{A}_{21}$ is an (n-m)xm matrix, $\underline{A}_{22}$ is an (n-m)x(n-m) matrix, $\underline{B}_1$ is an mxℓ matrix, and $\underline{B}_2$ is an (n-m)xℓ matrix. Therefore, $\hat{\underline{x}}(t)$, the estimate for $\tilde{\underline{x}}(t)$ (which is, for the case $\underline{C} = [\ \underline{I} \vdots \underline{0}\ ]$ and $\underline{D} = [\ \underline{0} \vdots \underline{I}\ ]$, the estimate for $\underline{x}(t)$), becomes

$$\hat{\underline{x}}(t) = \left[\begin{array}{c} \underline{y}(t) \\ \hline \underline{z}(t) + \underline{G}\underline{y}(t) \end{array}\right] .$$

Defining the error in the estimate $\hat{\underline{w}}(t)$ as

$$\underline{e}(t) \triangleq \hat{\underline{w}}(t) - \underline{w}(t)$$

and substituting (15) and (17) for $\underline{w}(t)$ and $\hat{\underline{w}}(t)$, respectively, yields

$$\underline{e}(t) = \underline{z}(t) + (\underline{G}\underline{C}-\underline{D})\underline{x}(t).$$

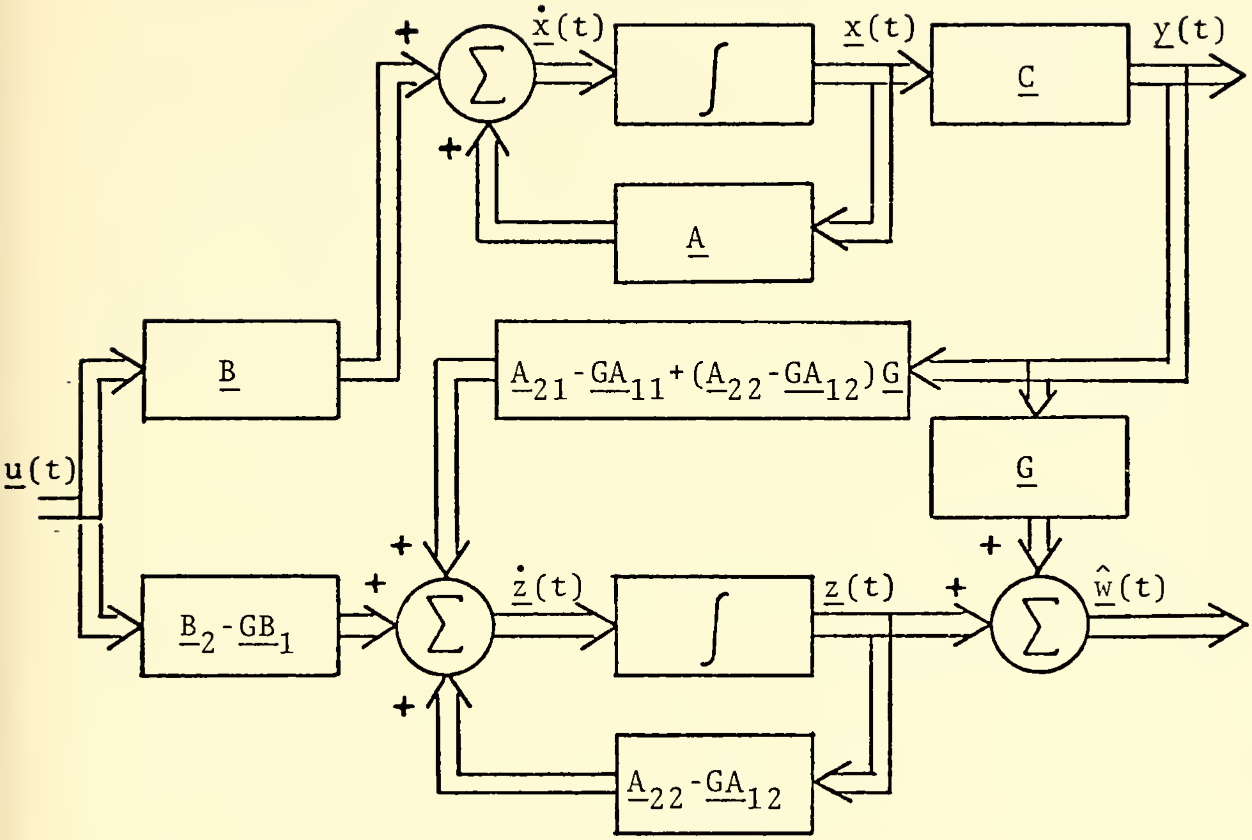


Figure 2. Reduced-Order Observer.

Differentiating with respect to time, substituting (14) and (16) for $\dot{\underline{x}}(t)$ and $\dot{\underline{z}}(t)$, respectively, and solving for $\dot{\underline{e}}(t)$ yields

$$\dot{\underline{e}}(t) = (\underline{A}_{22}-\underline{GA}_{12})\underline{z}(t) + [\underline{A}_{21}-\underline{GA}_{11} + (\underline{A}_{22}-\underline{GA}_{12})\underline{G}]\underline{y}(t)$$

$$+ (\underline{B}_2-\underline{GB}_1)\underline{u}(t) + (\underline{GC}-\underline{D})(\underline{Ax}(t) + \underline{Bu}(t)).$$

We note that for C = $[\ \underline{I} \ \vdots\ \underline{0}\ ]$ and D = $[\ \underline{0}\ \vdots\ \underline{I}\ ]$

$$\begin{aligned} \underline{GCAx}(t) &= [\underline{GA}_{11} \ \vdots\ \underline{GA}_{12}]\underline{x}(t) \\ -\underline{DAx}(t) &= [-\underline{A}_{21} \ \vdots\ -\underline{A}_{22}]\underline{x}(t) \\ \underline{GCBu}(t) &= \underline{GB}_1\underline{u}(t) \\ -\underline{DBu}(t) &= -\underline{B}_2\underline{u}(t). \end{aligned}$$

We note also that for $\underline{y}(t) = \underline{Cx}(t)$

$$\begin{aligned} \underline{A}_{21}\underline{y}(t) &= [\underline{A}_{21} \ \vdots\ \underline{0}]\underline{x}(t) \\ -\underline{GA}_{11}\underline{y}(t) &= [-\underline{GA}_{11} \ \vdots\ \underline{0}]\underline{x}(t). \end{aligned}$$

These relationships, when substituted into the above expression for $\dot{\underline{e}}(t)$, yield

$$\dot{\underline{e}}(t) = (\underline{A}_{22}-\underline{GA}_{12})\underline{z}(t) + (\underline{A}_{22}-\underline{GA}_{12})\underline{GCx}(t)$$

$$-[\ \underline{0}\ \vdots\ (\underline{A}_{22}-\underline{GA}_{12})]\underline{x}(t).$$

We note that the last term in this expression can be written as

$$-\ [\ \underline{0}\ \vdots\ (\underline{A}_{22}-\underline{GA}_{12})]\underline{x}(t) = -\ (\underline{A}_{22}-\underline{GA}_{12})\underline{Dx}(t).$$

Substituting this relationship into the expression for $\dot{\underline{e}}(t)$ and combining terms yields

$$\dot{\underline{e}}(t) = (\underline{A}_{22}-\underline{GA}_{12})(\underline{z}(t) + (\underline{GC}-\underline{D})\underline{x}(t))$$

where

$$\underline{z}(t) + (\underline{GC}-\underline{D})\underline{x}(t) = \underline{e}(t).$$

Therefore $\dot{\underline{e}}(t)$ becomes

$$\dot{\underline{e}}(t) = (\underline{A}_{22}-\underline{GA}_{12})\underline{e}(t).$$

If we define

$$\underline{F} \triangleq (\underline{A}_{22}-\underline{GA}_{12}) \tag{18}$$

then

$$\dot{\underline{e}}(t) = \underline{Fe}(t). \tag{19}$$

Defining the quadratic performance measure as

$$J \triangleq \int_0^\infty t^r \underline{e}^T(t)\underline{Qe}(t)dt \tag{20}$$

we see that the system (19) and the performance measure (20) have the same form as (1) and (2), respectively, and that (20) can therefore be written as

$$J = (-1)^{r+1} r!\underline{e}^T(0)\underline{V}_{r+1}\underline{e}(0).$$

By using (6) we have

$$\begin{aligned} \underline{F}^T\underline{V}_{s+1} + \underline{V}_{s+1}\underline{F} &= \underline{V}_s \qquad s = 1,2,\ldots,r \\ &\vdots \\ \underline{F}^T\underline{V}_1 + \underline{V}_1\underline{F} &= -\underline{Q}. \end{aligned}$$

Once again, the algorithm described in Section IID can be used to find $\underline{G}^*$, the observer gain matrix which is optimal with respect to the performance measure (20).

C. EXAMPLE OBSERVER DESIGN PROBLEM

To illustrate the application of the algorithm to observer design we consider the plant whose transfer function

is given by

$$\frac{Y(S)}{U(S)} = \frac{1}{S^3 + 12S^2 + 47S + 60} .$$

The state and output equations for this plant can be written as

$$\dot{\underline{x}}(t) = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -60 & -47 & -12 \end{bmatrix} \underline{x}(t) + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \underline{u}(t)$$

$$\underline{y}(t) = [\ 1\ 0\ 0\ ]\underline{x}(t).$$

Here,

$$\underline{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ -60 & -47 & -12 \end{bmatrix}, \quad \underline{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, \quad \underline{C} = [\ 1\ 0\ 0\ ].$$

It can be easily shown that $(\underline{C},\underline{A})$ is completely observable. A second-order observer with arbitrary eigenvalues can therefore be constructed to provide estimates for the unmeasured states $x_2(t)$ and $x_3(t)$. These estimates, $\hat{x}_2(t)$ and $\hat{x}_3(t)$, are given by

$$\begin{bmatrix} \hat{x}_2(t) \\ \hat{x}_3(t) \end{bmatrix} = \hat{\underline{w}}(t) = \underline{z}(t) + \underline{Gy}(t).$$

We require an initial estimate for the observer gain matrix $\underline{G}$ which yields a stable observer. The observer $\underline{F}$ matrix is given by (18) as

$$\underline{F} = \underline{A}_{22} - \underline{GA}_{12}. \tag{18}$$

Here

$$\underline{A}_{22} = \begin{bmatrix} 0 & 1 \\ -47 & -12 \end{bmatrix}, \quad \underline{A}_{12} = [\ 1\ 0\ ], \quad \underline{G} = \begin{bmatrix} g_1 \\ g_2 \end{bmatrix}.$$

Therefore

$$\underline{F} = \begin{bmatrix} -g_1 & 1 \\ -47-g_2 & -12 \end{bmatrix}$$

where the observer eigenvalues are the eigenvalues of $\underline{F}$ and are given by the solution to

$$\lambda^2 + (12+g_1)\lambda + (12g_1+g_2+47) = 0.$$

We arbitrarily select the initial estimate for the observer eigenvalues to be -7, -2. The characteristic equation is therefore

$$(\lambda+7)(\lambda+2) = 0$$

or

$$\lambda^2 + 9\lambda + 14 = 0.$$

Equating coefficients of the like powers of λ and solving for the $\underline{G}$ matrix yields

$$\underline{G} = \begin{bmatrix} -3 \\ 3 \end{bmatrix}.$$

This choice for $\underline{G}$ must also satisfy all design constraints. A parameter in observer design which must be limited is the observer speed since, as the observer gains are adjusted to cause the errors to die out more rapidly, the observer becomes more sensitive to noise. One measure of the observer

speed is the sum of its eigenvalues. It can be easily shown that this sum is numerically equal to the trace (the sum of the diagonal elements) of the $\underline{F}$ matrix.

As our constraint on observer speed we therefore set a limit, T, to the most negative value that we will allow the trace of the matrix $\underline{F}$ to assume. To observe the effect of T on the resulting design we will allow T to take on the values -9, -12, and -15. We see that for our initial estimate for $\underline{G}$ that the trace of $\underline{F}$ is -9 and that this $\underline{G}$ will therefore satisfy any of the values of T previously stated.

The design algorithm of Section IID was used to compute the optimal observer gain matrix G^*, the corresponding value of the performance measure $J(\underline{G}^*)$, the observer eigenvalues λ_1 and λ_2, and the worst case initial condition error vector for various choices of the design specifications $\underline{Q}$ and r in the performance measure (20) and for various choices of the constraint T. The plant and observer were then simulated on the IBM 360 digital computer for each design specification using the corresponding $\underline{G}^*$ and the worst case initial condition error.

Table I shows the effect of varying the design parameter r on $\underline{G}^*$, $J(\underline{G}^*)$, and the observer eigenvalues λ_1 and λ_2. For these cases the matrix $\underline{Q}$ is fixed at

$$\underline{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

and T is fixed at -12. Note that in each case, the value for g_1^* is zero. Since the trace $\underline{F}$ must be greater than or equal to T we see that

$$- 12 - g_1 \geq - 12$$

is satisfied only for g_1 less than or equal to zero. The fact that g_1^* is always zero indicates that the constraint is always active.

The effect of varying the parameter r on the observer error time response is shown in Fig. 3. Note that as the value of r is increased, the estimate error for both states reaches zero sooner. For the estimate error for the state $x_3(t)$, however, this is at the expense of larger error values in the early part of the operating interval.

In Table II the effect of the design parameter q_{11} on $\underline{G}^*$, $J(\underline{G}^*)$, λ_1 and λ_2 is shown. Here, the value of r is fixed at zero and the value of T is fixed at -12. The matrix $\underline{Q}$ has the form

$$\underline{Q} = \begin{bmatrix} q_{11} & 0 \\ 0 & 1 \end{bmatrix} .$$

Again we note that the constraint T is active.

The observer error time response is shown in Fig. 4. The effect of increasing q_{11} on the plant states is similar to that of increasing r, although somewhat less dramatic.

In Table III the effect of the design parameter q_{22} on $\underline{G}^*$, $J(\underline{G}^*)$, λ_1, and λ_2 is shown. Again, the value of r is fixed at zero and the value of T is fixed at -12. The matrix $\underline{Q}$ has the form

$$\underline{Q} = \begin{bmatrix} 1 & 0 \\ 0 & q_{22} \end{bmatrix} .$$

r	0	1	2
g_1^*	0.0	0.0	0.0
g_2^*	-35.0	-19.4	27.5
$J(\underline{G}^*)$	1.04	0.40	0.11
λ_1	-10.9	- 8.9	-6.0-j6.2
λ_2	- 1.1	- 3.1	-6.0+j6.2

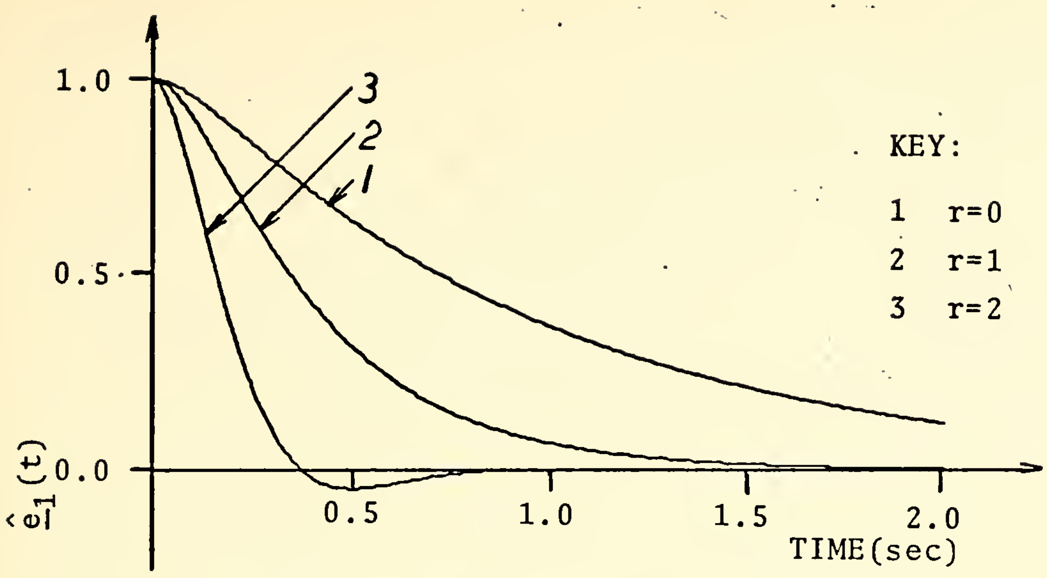
Table I. The effect of r on $\underline{G}^*$, $J(\underline{G}^*)$, λ_1 and λ_2.

$$\underline{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad T = -12$$

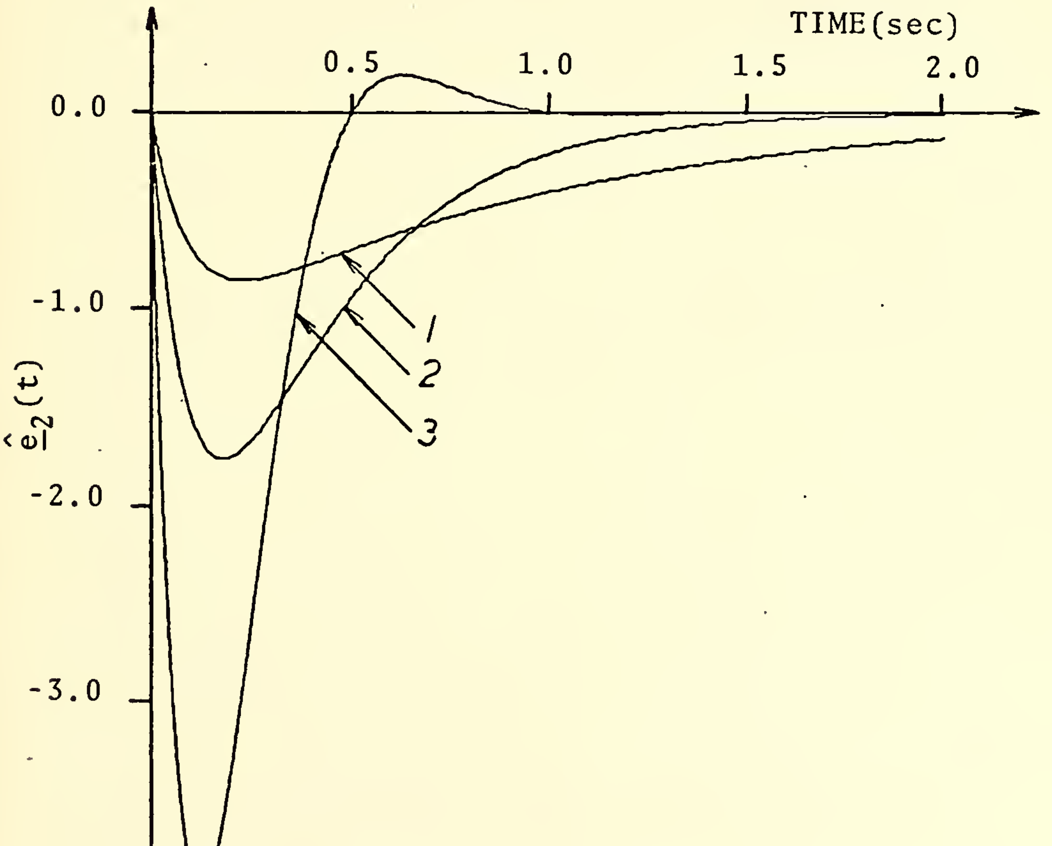
q_{11}	1	2	10
g_1^*	0.0	0.0	0.0
g_2^*	-35.0	-30.0	-8.9
$J(\underline{G}^*)$	1.04	1.50	3.58
λ_1	-10.9	-10.4	-6.0-j1.4
λ_2	- 1.1	- 1.6	-6.0+j1.4

Table II. The effect of q_{11} on $\underline{G}^*$, $J(\underline{G}^*)$, λ_1 and λ_2.

$r = 0$, $T = -12$

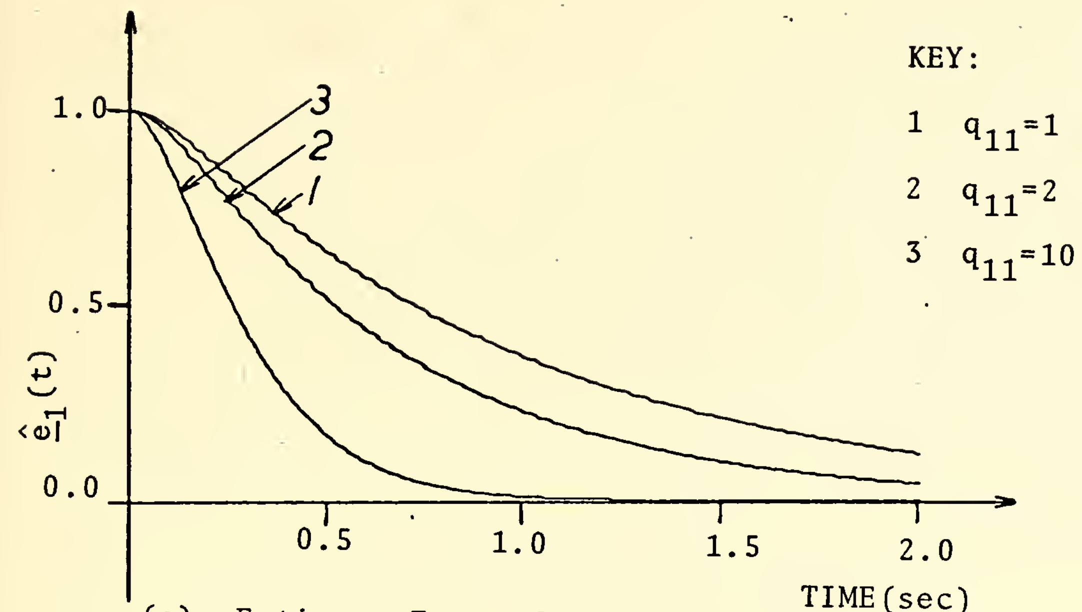


(a) Estimate Error for Plant State $x_2(t)$

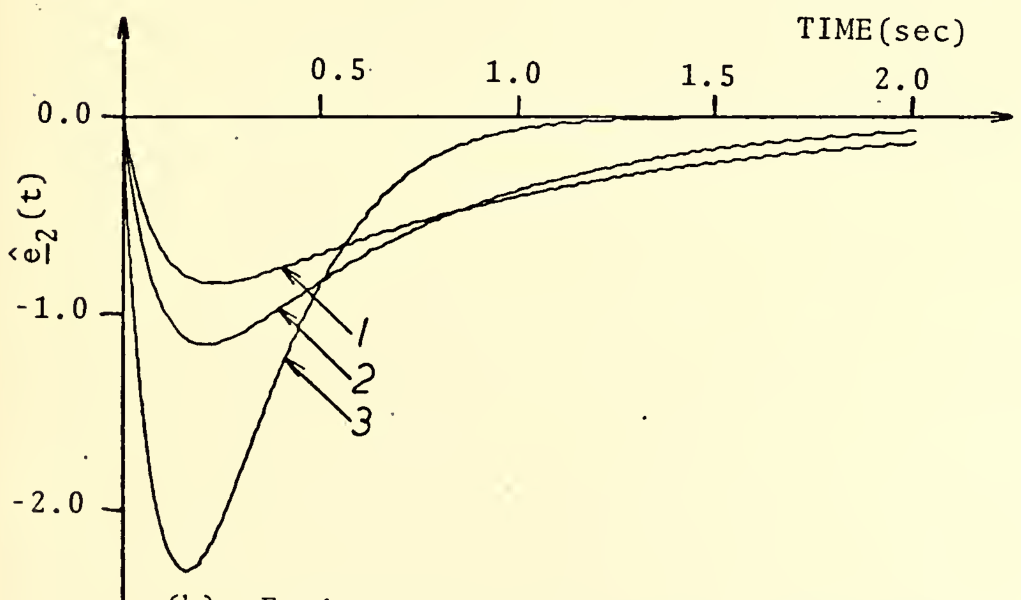


(b) Estimate Error for Plant State $x_3(t)$

Figure 3. Observer Estimate Error Time Response as a Function of Time Multiplier Exponent r.



(a) Estimate Error for Plant State $x_2(t)$



(b) Estimate Error for Plant State $x_3(t)$

Figure 4. Observer Estimate Error Time Response as a Function of q_{11}.

Once again we see that the constraint T is active.

The observer error time response is shown in Fig. 5. The effect of q_{22} on the response is quite different from the effects of r and q_{11} as we might expect. By placing greater emphasis on the cost associated with states which represent higher derivatives we do not allow them to deviate as far from zero and thus tend to slow the convergence process down. Hence, as q_{22} increases, the observer error states both reach zero at later values of time.

It should be pointed out that all observer configurations were simulated for a period of two seconds for the sake of uniformity in the graphical presentations. If the response of Fig. 5 were continued for a greater length of time, the curves would all eventually approach zero.

In Table IV the effect of the design constraint T on $\underline{G}^*$, $J(\underline{G}^*)$, λ_1 and λ_2 is shown. Again, r is fixed at zero and $\underline{Q}$ is fixed at

$$\underline{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} .$$

By noting the values for g_1^* we see that the constraint T is always active.

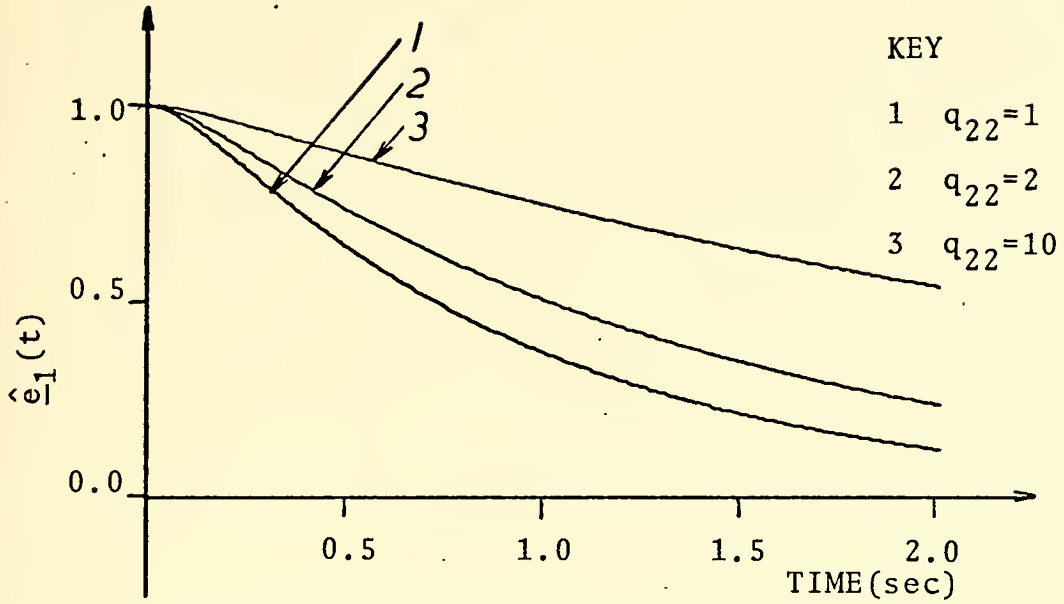
The observer error time response is shown in Fig. 6. We see that the effect of making T more negative is to improve the shape of the response at the earlier values of time. Note that in the case where T has the value -9 that the observer error in state $x_2(t)$ actually gets worse for early values of time.

q_{22}	1	2	10
g_1^*	0.0	0.0	0.0
g_2^*	-35.0	-38.5	-43.2
$J(\underline{G}^*)$	1.04	1.46	3.21
λ_1	-10.9	-11.2	-11.7
λ_2	- 1.1	- 0.8	- 0.3

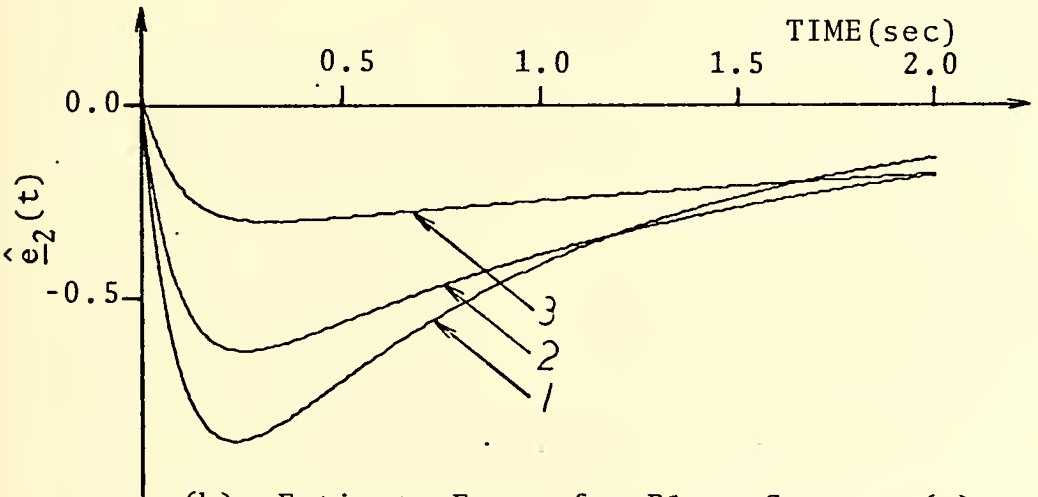
Table III. The effect of q_{22} on $\underline{G}^*$, $J(\underline{G}^*)$, λ_1 and λ_2.
$r = 0$, $T = -12$

T	-9	-12	-15
g_1^*	-3.0	0.0	3.0
g_2^*	27.1	-35.0	-44.9
$J(\underline{G}^*)$	8.30	1.04	0.16
λ_1	-4.5-j4.2	-10.9	-11.8
λ_2	-4.5+j4/2	- 1.1	- 3.2

Table IV. The effect of T on $\underline{G}^*$, $J(\underline{G}^*)$, λ_1 and λ_2.
$\underline{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$, $r = 0$.

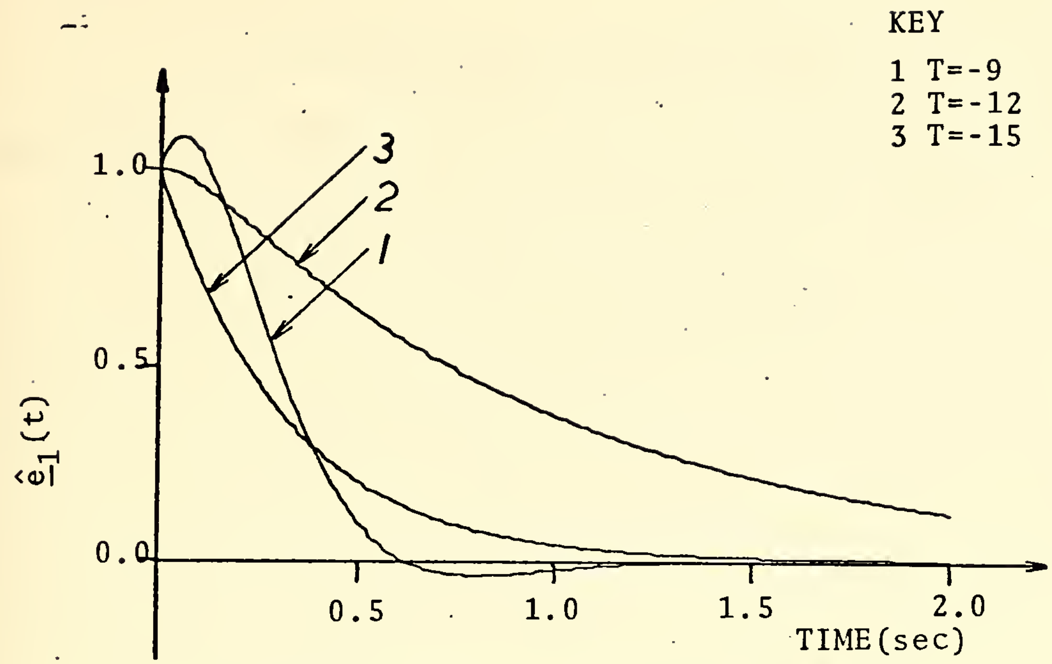


(a) Estimate Error for Plant State $x_2(t)$

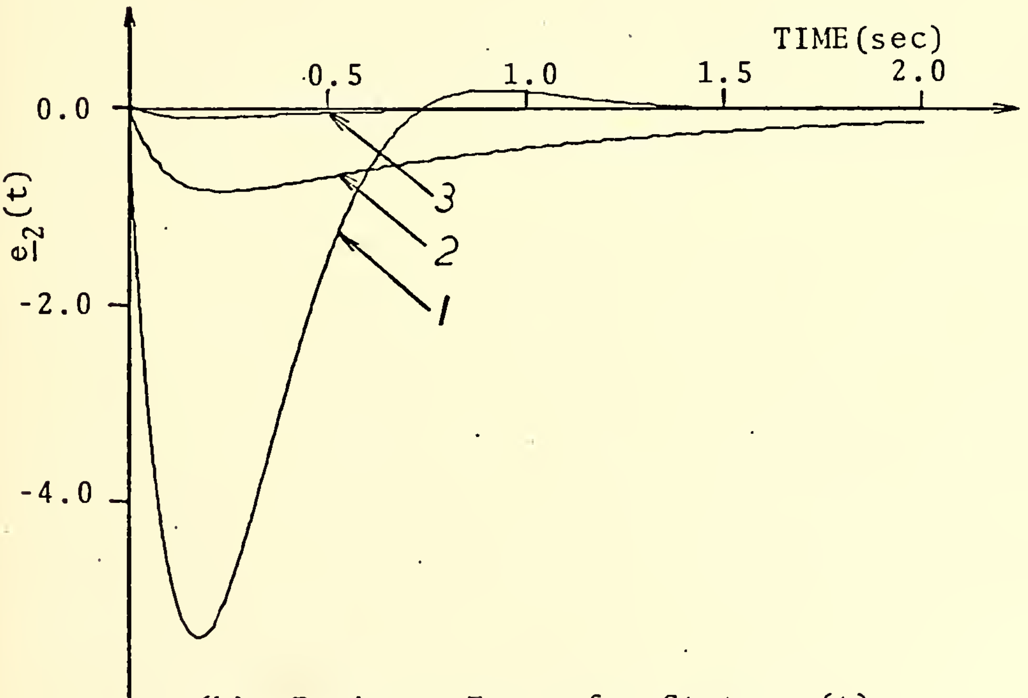


(b) Estimate Error for Plant State $x_3(t)$

Figure 5. Observer Estimate Error Time Response as a Function of q_{22}.



(a) Estimate Error for State $x_2(t)$



(b) Estimate Error for State $x_3(t)$

Figure 6. Observer Estimate Error Time Response as a Function of Constraint T.

IV. OBSERVER/CONTROLLER DESIGN

A. TRADITIONAL DESIGN TECHNIQUE

The traditional method for the design of feedback control systems utilizing observer estimates of unmeasured states is to specify desired plant performance in the manner described in Section IIA as if all states are available through direct measurements. A feedback network or controller is then constructed which causes the closed-loop system to satisfy these performance specifications. The controller has available as inputs either the observer estimate of the entire state vector or the measured plant states and the observer estimate of the remaining states. The observer is designed to be stable and to have a characteristic equation with eigenvalues which satisfy the somewhat nebulous requirement that the observer by "slightly faster than the plant" [2].

In this section, an alternative method is described in which the observer and controller are designed simultaneously using a performance measure similar in form to (2) in Section IIB. This results in a configuration in which both the controller design and the observer design are optimal with respect to the specified performance measure.

B. FULL-ORDER OBSERVER/CONTROLLER DESIGN

The full-order observer for the linear, time-invariant system

$$\dot{\underline{x}}(t) = \underline{Ax}(t) + \underline{Bu}(t)$$
$$\underline{y}(t) = \underline{Cx}(t) \tag{21}$$

is, as shown in Appendix D,

$$\dot{\underline{z}}(t) = (\underline{A}-\underline{GC})\underline{z}(t) + \underline{Gy}(t) + \underline{Bu}(t). \tag{22}$$

We define the control law as

$$\underline{u}(t) = \underline{Kx}(t)$$

where $\underline{K}$ is a constant ℓxn matrix.

It can be shown [2] that the stability of a closed-loop system using an observer output as a substitute for the plant states in the control law is unaffected by a stable observer. The control can therefore be approximated by the control estimate, $\hat{\underline{u}}(t)$, where

$$\hat{\underline{u}}(t) = \underline{Kz}(t). \tag{23}$$

The structure for this full-order observer/controller is shown in Fig. 7.

Defining the error state, $\underline{e}(t)$, as

$$\underline{e}(t) \triangleq \underline{z}(t) - \underline{x}(t) \tag{24}$$

and differentiating with respect to time yields

$$\dot{\underline{e}}(t) = \dot{\underline{z}}(t) - \dot{\underline{x}}(t).$$

Substituting (21) and (22) for $\dot{\underline{x}}(t)$ and $\dot{\underline{z}}(t)$, respectively, and solving for $\dot{\underline{e}}(t)$ yields

$$\dot{\underline{e}}(t) = (\underline{A}-\underline{GC})\underline{e}(t). \tag{25}$$

Substituting (23) into (21) yields

$$\dot{\underline{x}}(t) = \underline{Ax}(t) + \underline{BKz}(t). \tag{26}$$

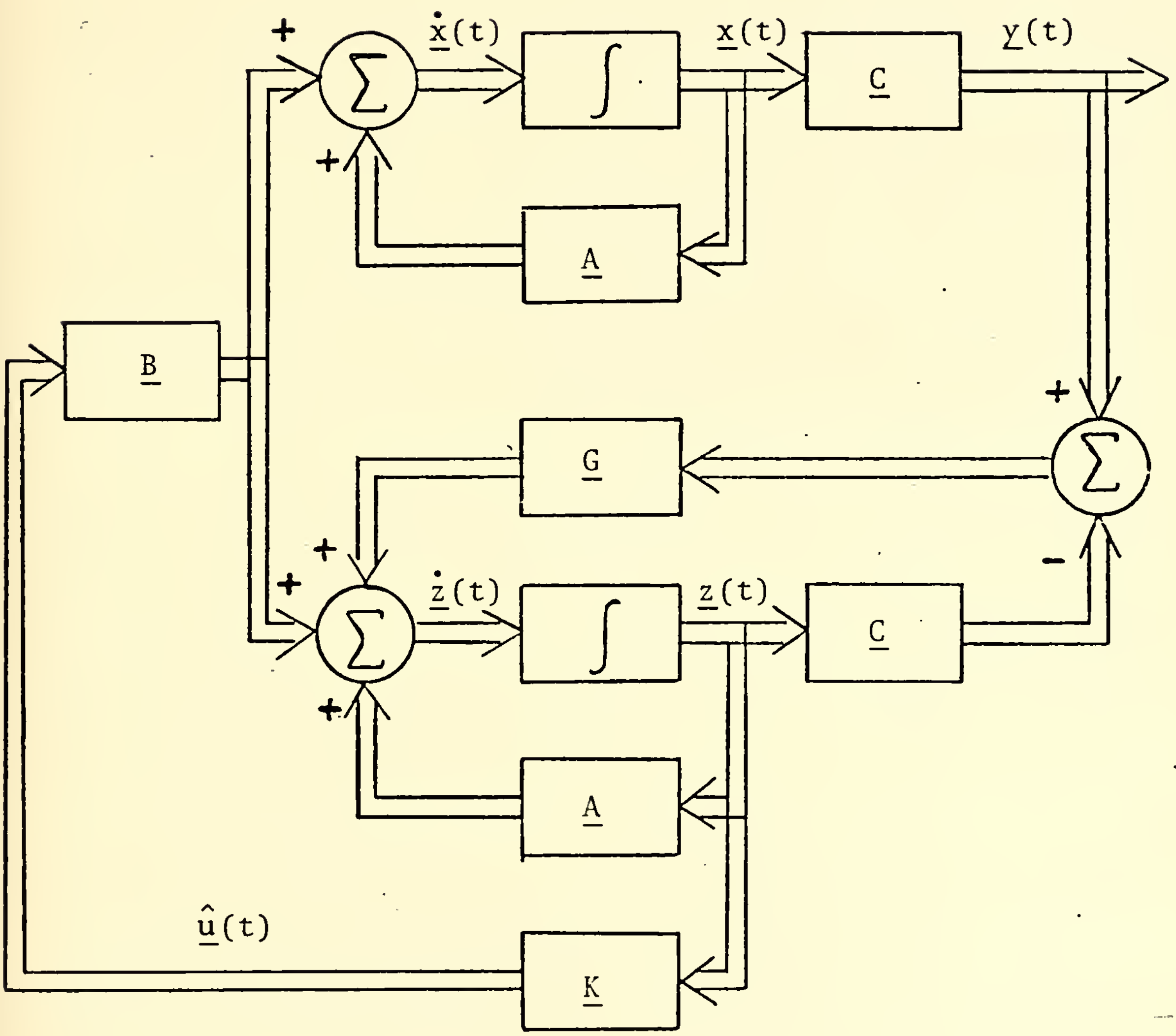


Figure 7. Full-Order Observer/Controller.

Solving (24) for $\underline{z}(t)$ and substituting into (26) yields

$$\dot{\underline{x}}(t) = (\underline{A}+\underline{BK})\underline{x}(t) + \underline{BKe}(t). \tag{27}$$

Defining the augmented state vector, $\underline{\xi}(t)$, as

$$\underline{\xi}(t) \triangleq \begin{bmatrix} \underline{x}(t) \\ \hline \underline{e}(t) \end{bmatrix}$$

and using (25) and (27), it is easily shown that

$$\dot{\underline{\xi}}(t) = \begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK} \\ \underline{0} & \underline{A}-\underline{GC} \end{bmatrix} \underline{\xi}(t) . \tag{28}$$

The performance measure can be defined as

$$J = \int_0^\infty t^r[\underline{x}^T(t)\underline{Qx}(t) + \underline{e}^T(t)\underline{Re}(t)]dt. \tag{29}$$

This performance measure is chosen because it accommodates the specification of convergence properties of both the closed-loop system states and the observer states and implicitly limits the expenditure of control effort. Equation (29) can be written in terms of $\underline{\xi}(t)$ as

$$J = \int_0^\infty t^r\underline{\xi}^T(t) \begin{bmatrix} \underline{Q} & \underline{0} \\ \underline{0} & \underline{R} \end{bmatrix} \underline{\xi}(t)dt. \tag{30}$$

Equations (28) and (30) are in the same form as (1) and (2), respectively. It has been shown [2] that the matrix $\underline{\Psi}$, defined as

$$\underline{\Psi} \triangleq \begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK} \\ \underline{0} & A-GC \end{bmatrix}$$

has as its eigenvalues, the eigenvalues of $\underline{A}+\underline{BK}$ and the eigenvalues of $\underline{A}-\underline{GC}$. The matrix $\underline{\Psi}$ is therefore a stability matrix if both $\underline{A}+\underline{BK}$ and $\underline{A}-\underline{GC}$ are stability matrices. It can be shown that the matrix

$$\begin{bmatrix} \underline{Q} & \underline{0} \\ \underline{0} & \underline{R} \end{bmatrix}$$

is positive definite if both $\underline{Q}$ and $\underline{R}$ are positive definite [7]. Therefore (30) can be rewritten as

$$J = (-1)^{r+1} r! \underline{\xi}^T(0) \underline{V}_{r+1} \underline{\xi}(0) \tag{31}$$

where, by using (6)

$$\begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK} \\ \underline{0} & \underline{A}-\underline{GC} \end{bmatrix}^T \underline{V}_{s+1} + \underline{V}_{s+1} \begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK} \\ \underline{0} & \underline{A}-\underline{GC} \end{bmatrix} = \underline{V}_s \qquad s=1,2,\ldots,r$$

$$\vdots$$

$$\begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK} \\ \underline{0} & \underline{A}-\underline{GC} \end{bmatrix}^T \underline{V}_1 + \underline{V}_1 \begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK} \\ \underline{0} & \underline{A}-\underline{GC} \end{bmatrix} = - \begin{bmatrix} \underline{Q} & \underline{0} \\ \underline{0} & \underline{R} \end{bmatrix} .$$

Furthermore, the algorithm described in Section IID can be applied to find $\underline{K}^*$ and $\underline{G}^*$, the feedback and observer gain matrices which are optimal with respect to the performance measure (29).

A slight modification to the design algorithm causes a significant reduction in the number of computations required on each iteration. To see why, we note that if the observer estimate initial condition vector is assumed to be the null vector, that is

$$\underline{z}(0) = \underline{0}$$

then, from (23), $\underline{e}(0)$ becomes

$$\underline{e}(0) = -\underline{x}(0)$$

and the augmented state vector $\underline{\xi}(0)$ becomes

$$\underline{\xi}(0) = \begin{bmatrix} \underline{x}(0) \\ ---- \\ -\underline{x}(0) \end{bmatrix} . \tag{32}$$

Partitioning the symmetric matrix $\underline{V}_{r+1}$ as

$$\underline{V}_{r+1} = \begin{bmatrix} \underline{V}_{11} & \underline{V}_{12} \\ \underline{V}_{12}^{T} & \underline{V}_{22} \end{bmatrix}$$

where each element is an nxn matrix, then substituting (32) and (33) into (31) yields

$$J = (-1)^{r+1} r! \underline{x}^{T}(0)[\underline{V}_{11} - \underline{V}_{12} - \underline{V}_{12}^{T} + \underline{V}_{22}]\underline{x}(0). \tag{34}$$

We now define the nxn matrix $\underline{V}'_{r+1}$ as

$$\underline{V}'_{r+1} = [\underline{V}_{11} - \underline{V}_{12} - \underline{V}_{12}^{T} + \underline{V}_{22}]. \tag{35}$$

Therefore

$$J = (-1)^{r+1} r! \underline{x}^{T}(0)\underline{V}'_{r+1}\underline{x}(0). \tag{36}$$

The matrix $\underline{V}'_{r+1}$ has n eigenvalues, while the matrix $\underline{V}_{r+1}$ has 2n eigenvalues. We see, therefore, that the effort required in finding J_1 can be significantly reduced by using (35) and (36) in step three of the design procedure.

C. REDUCED-ORDER OBSERVER/CONTROLLER DESIGN

For the linear, time-invariant system

$$\begin{aligned} \dot{\underline{x}}(t) &= \underline{Ax}(t) + \underline{Bu}(t) \\ \underline{y}(t) &= \underline{Cx}(t), \end{aligned} \tag{37}$$

we define the control law

$$\underline{u}(t) \triangleq \underline{Kx}(t) \tag{38}$$

where K is a constant ℓxn matrix. We define the vector

$$\tilde{\underline{x}}(t) \triangleq \begin{bmatrix} \underline{y}(t) \\ \hline \underline{w}(t) \end{bmatrix}$$

where $\underline{w}(t)$ is an (n-m)x1 vector. We let

$$\underline{w}(t) = \underline{Dx}(t) \tag{39}$$

where $\underline{D}$ is an (n-m)xn matrix, selected so that

$$\begin{bmatrix} \underline{C} \\ \hline \underline{D} \end{bmatrix}^{-1}$$

exists. By proper selection of the state variables, $\underline{C}$ will have the form

$$\underline{C} = [\ \underline{I} \vdots \underline{0}\ ].$$

A notationally convenient choice for $\underline{D}$ is

$$\underline{D} = [\ \underline{0} \vdots \underline{I}\ ].$$

For this choice of $\underline{C}$ and $\underline{D}$, $\hat{\underline{w}}(t)$, the estimate for $\underline{w}(t)$, can be obtained by constructing a reduced-order observer, as developed in Appendix D, whose state equations are given by

$$\dot{\underline{z}}(t) = (\underline{A}_{22}-\underline{G}\underline{A}_{12})\underline{z}(t) + (\underline{A}_{21}-\underline{G}\underline{A}_{11}+(\underline{A}_{22}-\underline{G}\underline{A}_{12})\underline{G})\underline{y}(t) + (\underline{B}_2-\underline{G}\underline{B}_1)\underline{u}(t) \tag{40}$$

where

$$\hat{\underline{w}}(t) = \underline{z}(t) + \underline{G}\underline{y}(t). \tag{41}$$

The matrix $\underline{K}$ and the vector $\underline{x}(t)$ can be partitioned as

$$\underline{K} = [\underline{K}^m : \underline{K}^o] \quad , \quad \underline{x}(t) = \begin{bmatrix} \underline{x}^m(t) \\ \hdashline \underline{x}^o(t) \end{bmatrix}$$

where, for the stated choice for $\underline{C}$ and $\underline{D}$,

$$\underline{y}(t) = \underline{x}^m(t), \; \underline{w}(t) = \underline{x}^o(t), \; \hat{\underline{w}}(t) = \hat{\underline{x}}^o(t).$$

The control law, given by (38), can therefore be approximated by

$$\hat{\underline{u}}(t) = \underline{K} \begin{bmatrix} \underline{x}^m(t) \\ \hdashline \hat{\underline{x}}^o(t) \end{bmatrix}$$

or

$$\hat{\underline{u}}(t) = [\underline{K}^m : \underline{K}^o] \begin{bmatrix} \underline{y}(t) \\ \hdashline \underline{z}(t) + \underline{G}\underline{y}(t) \end{bmatrix}$$

or

$$\hat{\underline{u}}(t) = \underline{K}^m\underline{y}(t) + \underline{K}^o\underline{z}(t) + \underline{K}^o\underline{G}\underline{y}(t). \tag{42}$$

The structure for this reduced-order observer/controller is shown in Fig. 8.

We define the observation error, $\underline{e}(t)$, as

$$\underline{e}(t) \triangleq \hat{\underline{w}}(t) - \underline{w}(t). \tag{43}$$

Substituting (41) for $\hat{\underline{w}}(t)$ and (39) for $\underline{w}(t)$ yields

$$\underline{e}(t) = \underline{z}(t) + \underline{G}\underline{y}(t) - \underline{D}\underline{x}(t). \tag{44}$$

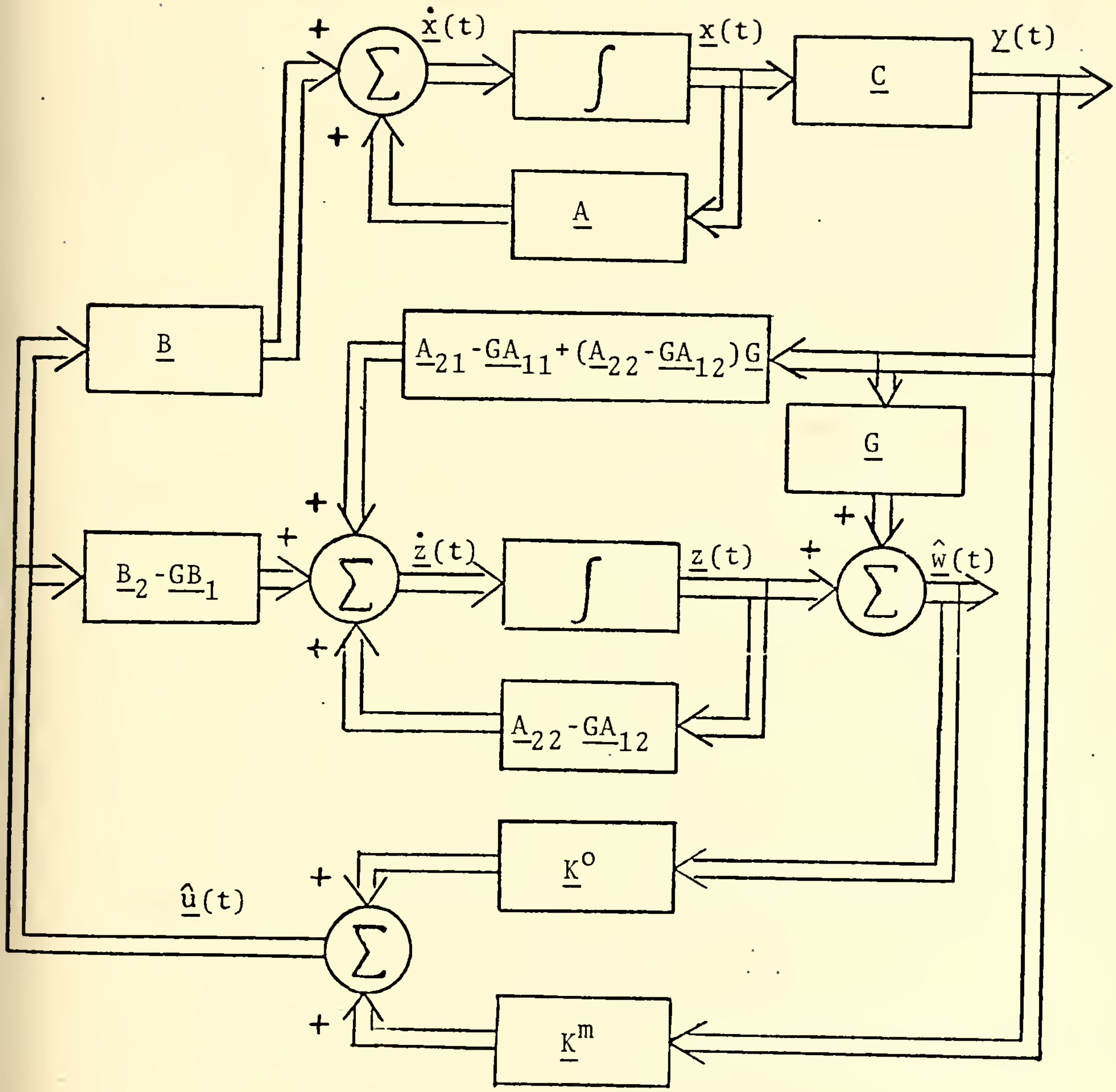


Figure 8. Reduced-Order Observer/Controller.

Substituting (37) for $\underline{y}(t)$ and rearranging yields

$$\underline{e}(t) = \underline{z}(t) + (\underline{GC}-\underline{D})\underline{x}(t).$$

Differentiating with respect to time yields

$$\dot{\underline{e}}(t) = \dot{\underline{z}}(t) + (\underline{GC}-\underline{D})\dot{\underline{x}}(t).$$

Substituting (37) and (40) for $\dot{\underline{x}}(t)$ and $\dot{\underline{z}}(t)$, respectively, and solving for $\dot{\underline{e}}(t)$ yields, as shown in Section IIIB,

$$\dot{\underline{e}}(t) = (\underline{A}_{22}-\underline{GA}_{12})\underline{e}(t).$$

Solving (44) for $\underline{z}(t) + \underline{Gy}(t)$ and substituting into (42) yields

$$\hat{\underline{u}}(t) = \underline{K}^m\underline{y}(t) + \underline{K}^o(\underline{e}(t) + \underline{Dx}(t)).$$

We note, however, that

$$\underline{Dx}(t) = \underline{x}^o(t)$$

and

$$\underline{y}(t) = \underline{x}^m(t).$$

Therefore,

$$\hat{\underline{u}}(t) = \underline{Kx}(t) + \underline{K}^o\underline{e}(t). \qquad (46)$$

Substituting this estimate for $\underline{u}(t)$ into (37) yields

$$\dot{\underline{x}}(t) = \underline{Ax}(t) + \underline{B}(\underline{Kx}(t) + \underline{K}^o\underline{e}(t)$$

which, after rearranging, becomes

$$\dot{\underline{x}}(t) = (\underline{A}+\underline{BK})\underline{x}(t) + \underline{BK}^o\underline{e}(t)). \qquad (47)$$

We define the augmented state vector, $\underline{\xi}(t)$, as

$$\underline{\xi}(t) \triangleq \begin{bmatrix} \underline{x}(t) \\ \hline \underline{e}(t) \end{bmatrix} .$$

Using this definition and the expressions for $\dot{\underline{x}}(t)$ and $\dot{\underline{e}}(t)$ given by (47) and (45), respectively, yields

$$\dot{\underline{\xi}}(t) = \begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK}^o \\ \underline{0} & \underline{A}_{22}-\underline{GA}_{12} \end{bmatrix} \underline{\xi}(t) . \tag{48}$$

The performance measure can be defined as

$$J = \int_0^\infty t^r[\underline{x}^T(t)\underline{Q}\underline{x}(t) + \underline{e}^T(t)\underline{R}\underline{e}(t)]dt \tag{49}$$

which can be rewritten in terms of the augmented state vector, $\underline{\xi}(t)$, as

$$J = \int_0^\infty t^r \underline{\xi}^T(t) \begin{bmatrix} \underline{Q} & \underline{0} \\ \underline{0} & \underline{R} \end{bmatrix} \underline{\xi}(t)dt. \tag{50}$$

Equations (48) and (50) are in the same form as (1) and (2), respectively. Furthermore, the matrix

$$\begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK}^o \\ \underline{0} & \underline{A}_{22}-\underline{GA}_{12} \end{bmatrix}$$

is a stability matrix if both $\underline{A}+\underline{BK}$ and $\underline{A}_{22}-\underline{GA}_{12}$ are stability matrices. Also, the matrix

$$\begin{bmatrix} \underline{Q} & \underline{0} \\ \underline{0} & \underline{R} \end{bmatrix}$$

is positive definite if both $\underline{Q}$ and $\underline{R}$ are positive definite.

The performance measure (50) can therefore be rewritten as

$$J = (-1)^{r+1} r! \underline{\xi}^T(0) \underline{V}_{r+1} \underline{\xi}(0) \tag{51}$$

where, by using (6),

$$\begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK}^o \\ \underline{0} & \underline{A}_{22}-\underline{GA}_{12} \end{bmatrix}^T \underline{V}_{s+1} + \underline{V}_{s+1} \begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK}^o \\ \underline{0} & \underline{A}_{22}-\underline{GA}_{12} \end{bmatrix} = \underline{V}_s$$

$$s = 1,2,\ldots,r$$

$$\vdots$$

$$\begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK}^o \\ \underline{0} & \underline{A}_{22}-\underline{GA}_{12} \end{bmatrix}^T \underline{V}_1 + \underline{V}_1 \begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK}^o \\ \underline{0} & \underline{A}_{22}-\underline{GA}_{12} \end{bmatrix} = - \begin{bmatrix} \underline{Q} & \underline{0} \\ \underline{0} & \underline{R} \end{bmatrix}$$

Furthermore, the design procedure of Section IID can be applied to determine $\underline{K}^*$ and $\underline{G}^*$, the feedback and observer gain matrices which are optimal with respect to the specified performance measure (49).

Once again, it is possible to reduce the number of computations required for each iteration of the design algorithm. In this case, we set the observer estimate, $\hat{\underline{w}}(0) = \underline{0}$, which yields $\underline{e}(0) = -\underline{Dx}(0)$. Therefore $\underline{\xi}(0)$ becomes

$$\underline{\xi}(0) = \begin{bmatrix} \underline{x}(0) \\ \hline -\underline{Dx}(0) \end{bmatrix} .$$

Substituting this into (51) yields

$$J = (-1)^{r+1} r! \begin{bmatrix} \underline{x}(0) \\ \hline -\underline{Dx}(0) \end{bmatrix}^T \underline{V}_{r+1} \begin{bmatrix} \underline{x}(0) \\ \hline -\underline{Dx}(0) \end{bmatrix} . \tag{52}$$

Partitioning the (2n-m)x(2n-m) symmetric matrix $\underline{V}_{r+1}$ as

$$\underline{V}_{r+1} = \begin{bmatrix} \underline{V}_{11} & \underline{V}_{12} \\ \underline{V}_{12}^T & \underline{V}_{22} \end{bmatrix}$$

where $\underline{V}_{11}$ is an nxn matrix, $\underline{V}_{12}$ is an nx(n-m) matrix, and $\underline{V}_{22}$ is an (n-m)x(n-m) matrix and substituting into (52) yields, after expanding

$$J = (-1)^{r+1} r! \underline{x}^T(0)[\underline{V}_{11} - \underline{D}^T \underline{V}_{12}{}^T - \underline{V}_{12}\underline{D} + \underline{D}^T \underline{V}_{22}\underline{D}]\underline{x}(0). \tag{53}$$

We define the nxn matrix $\underline{V}'_{r+1}$ as

$$\underline{V}'_{r+1} \triangleq [\underline{V}_{11} - \underline{D}^T \underline{V}_{12}{}^T - \underline{V}_{12}D + \underline{D}^T \underline{V}_{22}\underline{D}]. \tag{54}$$

Therefore, (53) can be rewritten as

$$J = (-1)^{r+1} r! \underline{x}^T(0)\underline{V}'_{r+1}\underline{x}(0). \tag{55}$$

The matrix $\underline{V}'_{r+1}$ has n eigenvalues, while the matrix $\underline{V}_{r+1}$ has 2n-m eigenvalues. We see, therefore that the effort required in determining J_1 in step three of the design procedure can be reduced by using (54) and (55).

D. EXAMPLE OBSERVER/CONTROLLER DESIGN PROBLEM

To illustrate the application of the design algorithm to observer/controller design we consider the plant whose transfer function is given by

$$\frac{Y(S)}{U(S)} = \frac{1}{S^2+15S+50}.$$

The state and output equations for this plant can be written as

$$\dot{\underline{x}}(t) = \begin{bmatrix} 0 & 1 \\ -50 & -15 \end{bmatrix} \underline{x}(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \underline{u}(t)$$

$$\underline{y}(t) = [\ 1\ \ 0\ ]\underline{x}(t).$$

Here,

$$\underline{A} = \begin{bmatrix} 0 & 1 \\ -50 & -15 \end{bmatrix}, \quad \underline{B} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C = [\ 1\ \ 0\ ].$$

Then

$$\underline{A}+\underline{BK} = \begin{bmatrix} 0 & 1 \\ -50 & -15 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix}[k_1 \ \ k_2]$$

$$= \begin{bmatrix} 0 & 1 \\ k_1-50 & k_2-15 \end{bmatrix}.$$

Also

$$\underline{F} = \underline{A}_{22} - \underline{GA}_{12}$$

where

$$\underline{A}_{22} = [-15], \quad \underline{A}_{12} = [1], \quad \underline{G} = [g].$$

Therefore

$$\underline{F} = -[15+g].$$

We must select an initial estimate for both $\underline{K}$ and $\underline{G}$ which causes the matrices $\underline{A}+\underline{BK}$ and $\underline{F}$ to be stability matrices. As in the case of the observer alone we must constrain system speed. In this case, however, we sum the eigenvalues of the closed loop system. It is easily shown that this sum is given by the trace of the matrix

$$\begin{bmatrix} \underline{A}+\underline{BK} & \underline{BK}^{o} \\ 0 & \underline{A}_{22}-\underline{GA}_{12} \end{bmatrix}$$

which is given by

$$k_2 - 30 - g.$$

We arbitrarily select the trace constraint T to have a value of -25. Therefore

$$-25 \leq k_2 - 30 - g.$$

We see that the open-loop system is stable. We therefore choose an initial estimate for $\underline{K}$ of [0 0]. We must now choose a value for g which satisfies both

$$-25 \leq - 30 - g$$

and

$$-(15+g) < 0 .$$

Or, combining the two inequalities,

$$-15 \leq g < - 5.$$

We choose for our estimate of $\underline{G}$, therefore,

$$\underline{G} = [-10].$$

The design algorithm of Section IID was used to compute the optimal controller feedback gains $\underline{K}^*$ and observer gains $\underline{G}^*$, the corresponding value for the performance measure $J(\underline{K}^*,\underline{G}^*)$, the plant and observer eigenvalues λ_1, λ_2 and λ_3, and the worst case plant and observer estimate error initial conditions for various choices of $\underline{Q}$ and r in the performance measure (49). The constraint T was fixed at -25 and the design parameter $\underline{R}$ was fixed at

$$\underline{R} = [1]$$

in all cases. The plant and observer were simulated on the IBM 360 digital computer for each performance measure specified using $\underline{K}^*$, $\underline{G}^*$, and the corresponding plant and estimate worst case initial conditions.

Table V shows the effect of varying the design parameter r on $\underline{K}^*$, $\underline{G}^*$, $J(\underline{K}^*,\underline{G}^*)$, λ_1, λ_2, and λ_3. In this case the $\underline{Q}$ matrix is also fixed at

$$\underline{Q} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Figure 9 shows the effect of varying the design parameter r on the closed-loop system response to the worst case initial conditions. We see that increasing r improves the system settling time in both states at the expense of larger values of $x_2(t)$ in the early part of the operating interval.

Table VI shows the effect of varying the design parameter q_{11} on $\underline{K}^*$, $\underline{G}^*$, $J(\underline{K}^*,\underline{G}^*)$, λ_1, λ_2, and λ_3. In this case r is fixed at zero and $\underline{Q}$ has the form

$$\underline{Q} = \begin{bmatrix} q_{11} & 0 \\ 0 & 1 \end{bmatrix}.$$

Figure 10 shows the effect of varying q_{11} on the closed-loop system response to the worst case initial conditions. We see that as the value of q_{11} is increased, the settling time improves for both states at the expense of larger values of $x_2(t)$ in the early part of the operating interval.

Table VII shows the effect of varying the design parameter q_{22} on $\underline{K}^*$, $\underline{G}^*$, $J(\underline{K}^*,\underline{G}^*)$, λ_1, λ_2, and λ_3. In this case r is fixed at zero and $\underline{Q}$ has the form

$$\underline{Q} = \begin{bmatrix} 1 & 0 \\ 0 & q_{22} \end{bmatrix}.$$

r	0	1	2
k_1^*	25.7	-17.1	-217.4
k_2^*	- 9.4	- 9.0	- 8.0
g^*	-14.4	-14.0	-13.0
$J(\underline{K}^*,\underline{G}^*)$	1.02	0.34	0.06
λ_1	- 0.6	- 1.0	- 2.0
λ_2	-23.3	-20.8	-11.5-j11.6
λ_3	- 1.0	- 3.2	-11.5+j11.6

Table V. The effect of r on $\underline{K}^*$, $\underline{G}^*$, $J(\underline{K}^*,\underline{G}^*)$, λ_1, λ_2, and λ_3. $\underline{Q} = \begin{bmatrix}1 & 0\\ 0 & 1\end{bmatrix}$, $T = -25$, $\underline{R} = [1]$.

q_{11}	1	2	10
k_1^*	25.7	-15.5	-25.9
k_2^*	- 9.4	- 9.5	- 9.0
g^*	-14.4	-14.5	-14.0
$J(\underline{K}^*,\underline{G}^*)$	1.02	1.46	3.37
λ_1	- 0.6	- 0.5	- 1.0
λ_2	-23.3	-23.0	-20.3
λ_3	- 1.0	- 1.5	- 3.7

Table VI. The effect of q_{11} on $\underline{K}^*$, $\underline{G}^*$, $J(\underline{K}^*,\underline{G}^*)$, λ_1, λ_2, and λ_3. $r = 0$, $T = -25$, $\underline{R} = [1]$.

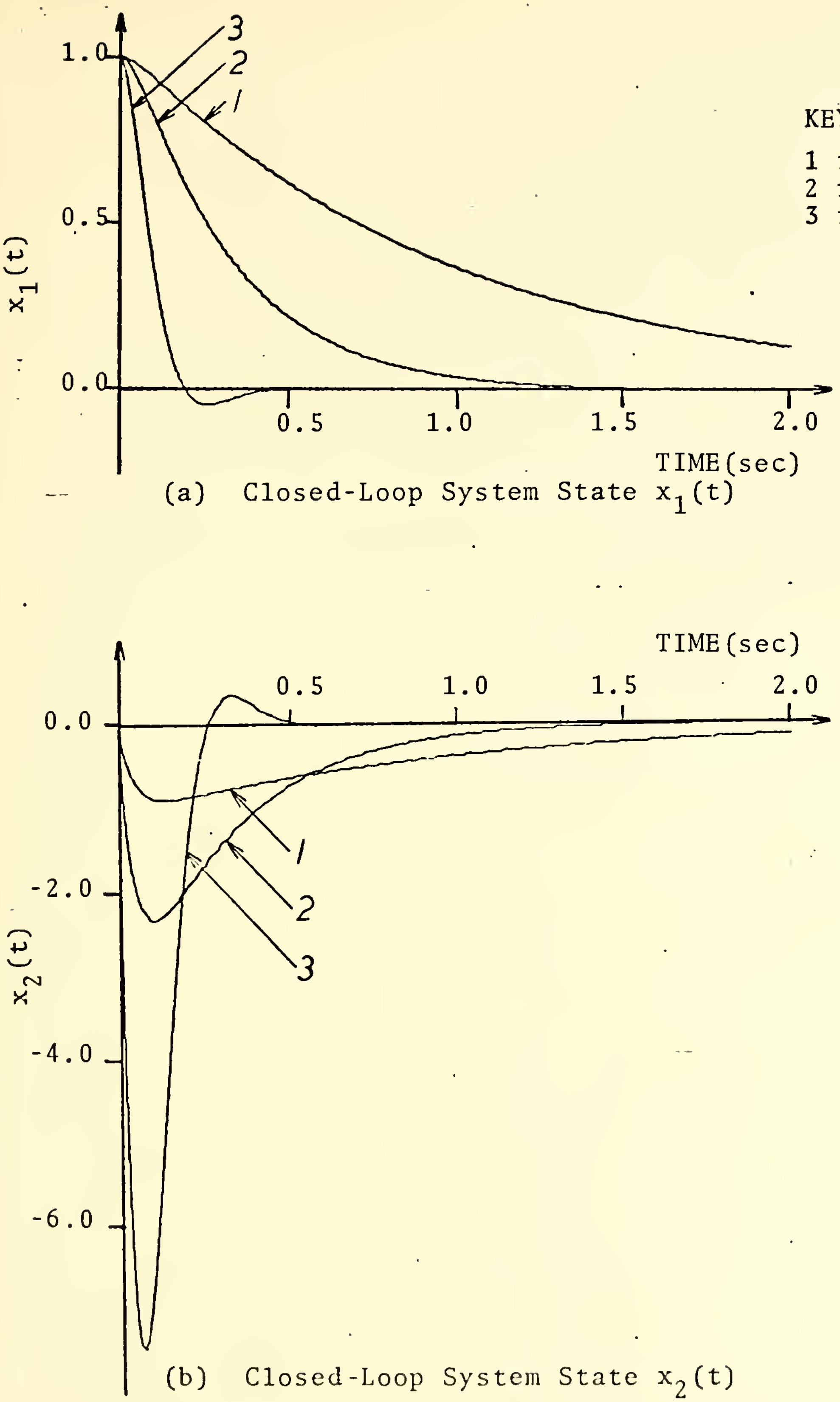
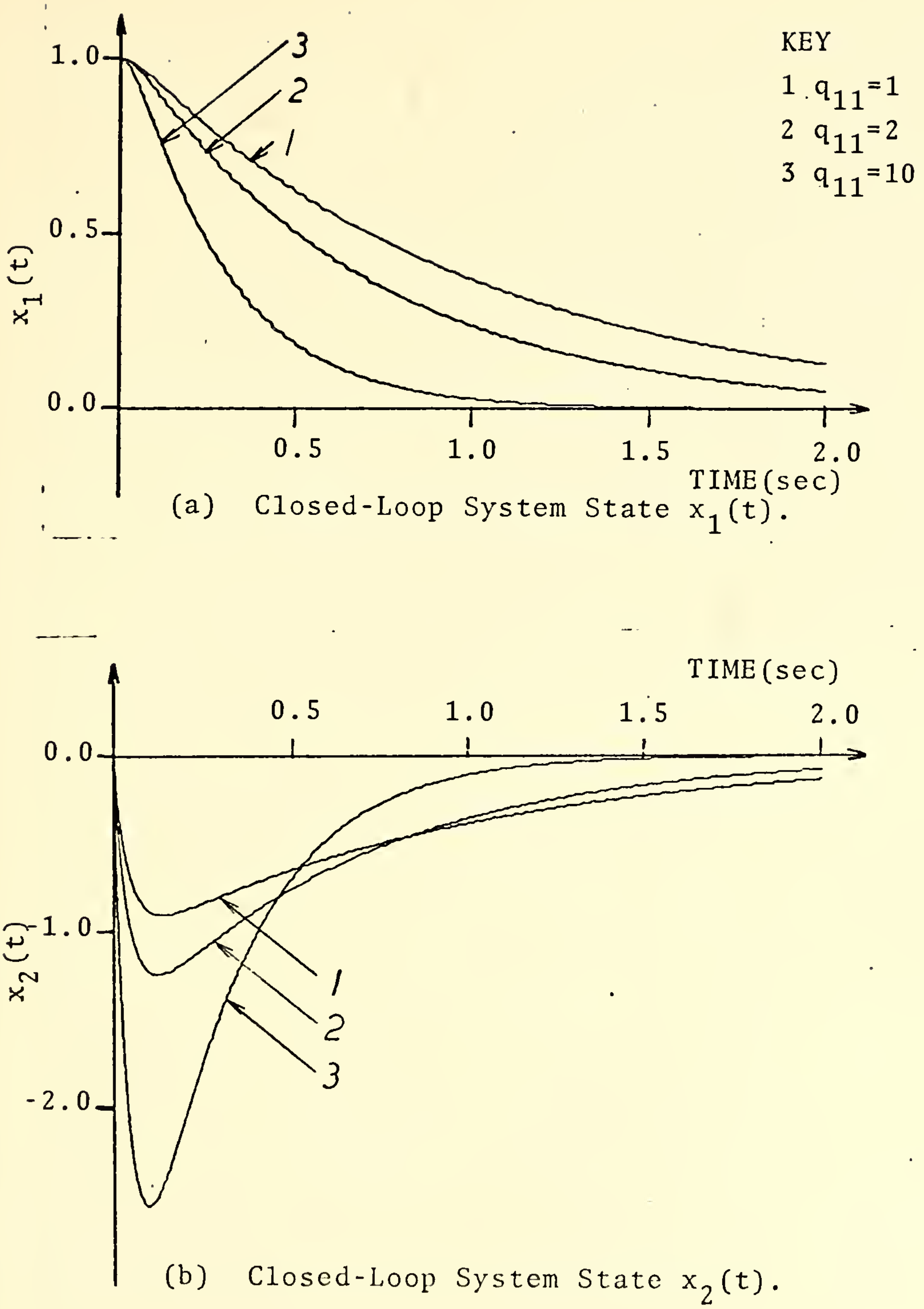


Figure 9. Closed-Loop System Time Response as a Function of Time Multiplier Exponent r.



(a) Closed-Loop System State $x_1(t)$.

(b) Closed-Loop System State $x_2(t)$.

Figure 10. Closed-Loop System Time Response as a Function of q_{11}.

q_{22}	1	2	10
k_1^*	25.7	32.7	42.2
k_2^*	- 9.4	- 9.4	- 9.6
g^*	-14.4	-14.4	-14.6
$J(\underline{K}^*,\underline{G}^*)$	1.02	1.44	3.18
λ_1	- 0.6	- 0.6	- 0.4
λ_2	-23.3	-23.7	-24.3
λ_3	- 1.0	- 0.7	- 0.3

Table VII. The effect of q_{22} on $\underline{K}^*, \underline{G}^*, J(\underline{K}^*, \underline{G}^*)$, λ_1, λ_2, and λ_3. $r=0$, $T=-25$, $\underline{R}=[1]$

Figure 11 shows the effect of varying q_{22} on the closed-loop system response to the worst case initial conditions. We see that as q_{22} is increased, the system becomes more sluggish, requiring longer settling times, as we would expect.

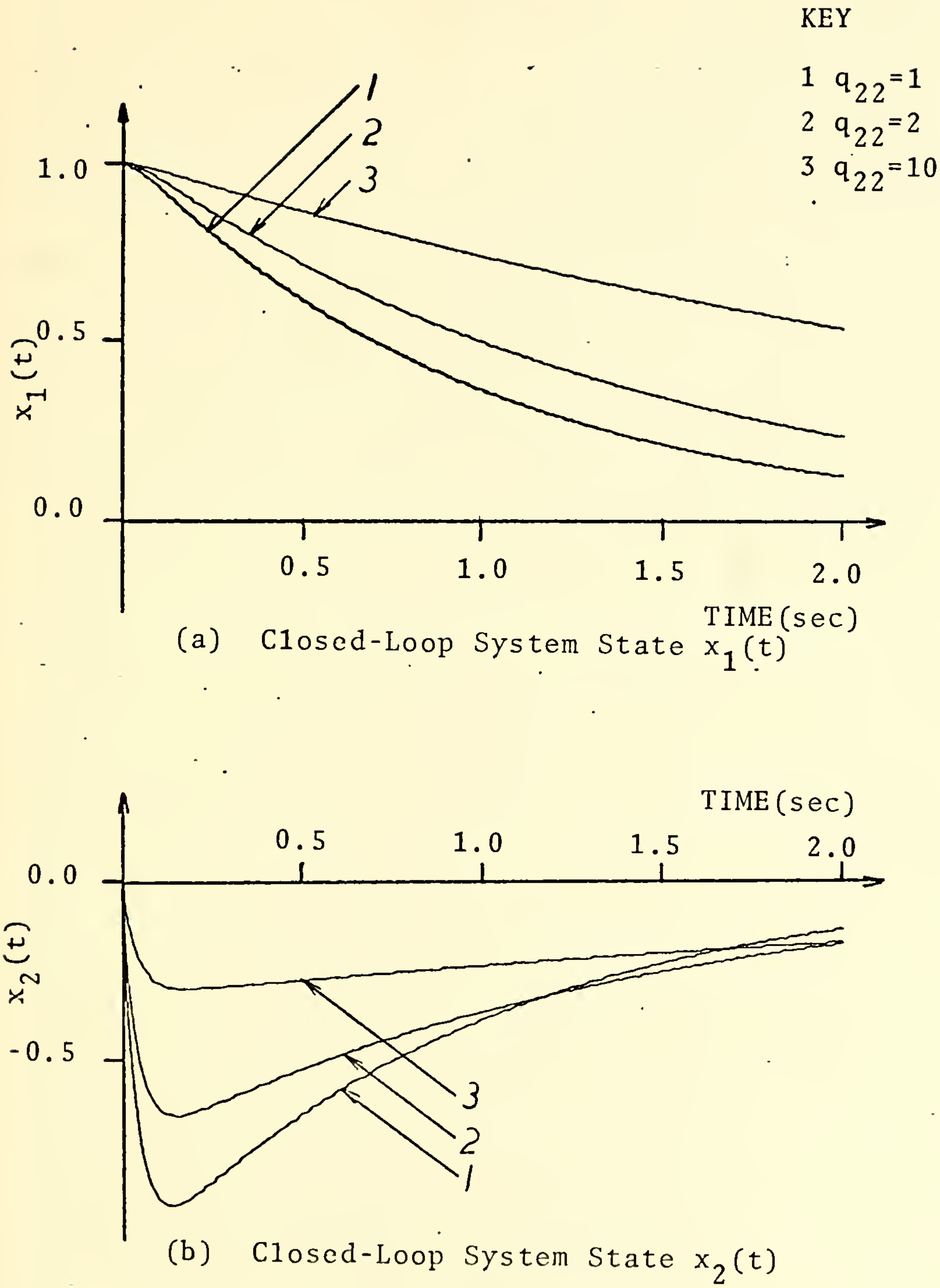


Figure 11. Closed-Loop System Time Response as a Function of q_{22}.

V. CONCLUSIONS

A new technique for observer and observer/controller design which requires the minimization of an integral of time-multiplied-square-error (ITSE) performance measure has been developed. An algorithm which is well suited to digital computation and which is especially useful for higher-order systems has been presented. An example of a computer program which implements the algorithm for the case of an observer/controller is shown in Appendix E. The computations are performed off-line with a typical computation time of thirty seconds.

It has been demonstrated through the use of examples that modification of the time response of an observer or a closed-loop system utilizing an observer/controller to initial condition error can be achieved by a suitable change in the appropriate design parameters of the ITSE performance measure. Furthermore, it has been demonstrated that the qualitative change in the system time response reflects directly the qualitative change in the performance measure design parameter.

It is recognized that for second-order and, perhaps, third-order single-input, single-output systems the relationships between traditional design specifications and system transfer functions are simple and relatively easy to use. In these cases, the design procedure suggested above is generally not efficient. However, when dealing with

higher-order, multiple-input, multiple-output systems, the advantages of the new technique become more significant. For these systems, the design specification and system transfer function relationships are highly complex and the traditional methods are therefore extremely difficult to use. However, the new technique offers a relatively straightforward approach to design which does not become more complex as the system order, the number of inputs, or the number of outputs increases.

APPENDIX A

EVALUATION OF THE INFINITE TIME INTEGRAL $\int_0^\infty t^r \underline{s}^T(t)\underline{Q}\underline{s}(t)dt$

For the system of linear, first-order differential equations

$$\dot{\underline{s}}(t) = \underline{H}\underline{s}(t) \tag{1A}$$

if there exists a scalar function

$$V_1(\underline{s}(t)) = \underline{s}^T(t)\underline{V}_1\underline{s}(t) \tag{2A}$$

subject to the conditions

1) $V_1(\underline{0}) = 0$

2) $V_1(\underline{s}(t)) > 0$, for all $\underline{s}(t) \neq \underline{0}$

3) $V_1(\underline{s}(t)) \rightarrow \infty$, as $||\underline{s}(t)|| \rightarrow \infty$

4) $\dot{V}_1(\underline{s}(t)) < 0$, for all $\underline{s}(t) \neq \underline{0}$

5) $\dot{V}_1(\underline{s}(t)) = 0$, $\underline{s}(t) = \underline{0}$

then the system (1A) is asymptotically stable [4].

Since $V_1(\underline{s}(t))$ is a scalar, it is equal to its transpose. Hence, from (2A), we see that the matrix $\underline{V}_1$ is symmetric. We now consider

$$\dot{V}_1(\underline{s}(t)) = \dot{\underline{s}}^T(t)\underline{V}_1\underline{s}(t) + \underline{s}^T(t)\underline{V}_1\dot{\underline{s}}(t).$$

Substituting (1A) for $\dot{\underline{s}}(t)$ yields

$$\dot{\underline{V}}_1(\underline{s}(t)) = \underline{s}^T(t)\underline{H}^T\underline{V}_1\underline{s}(t) + \underline{s}^T(t)\underline{V}_1\underline{H}\underline{s}(t)$$

which, after rearranging terms becomes

$$\dot{V}_1(\underline{s}(t)) = \underline{s}^T(t)[\underline{H}^T\underline{V}_1 + \underline{V}_1\underline{H}]\underline{s}(t).$$

We now define the matrix $\underline{Q}$ as

$$\underline{Q} \triangleq -[\underline{H}^T\underline{V}_1 + \underline{V}_1\underline{H}]$$

and note that since $\underline{V}_1$ is symmetric, then $\underline{Q}$ must also be symmetric. Therefore

$$\dot{V}_1(\underline{s}(t)) = -\underline{s}^T(t)\underline{Q}\underline{s}(t). \tag{3A}$$

At this point we note that equation (2A) insures that conditions one and three are satisfied. Condition two is satisfied if $\underline{V}_1$ is positive definite and conditions four and five are satisfied if $\underline{Q}$ is positive definite.

Integrating the left hand side of (3A) over the finite time interval, T, yields

$$\int_0^T \dot{V}_1(\underline{s}(t))dt = \int_0^T dV_1(\underline{s}(t))$$

$$\int_0^T \dot{V}_1(\underline{s}(t))dt = V_1(\underline{s}(T)) - V_1(\underline{s}(0)).$$

If (1A) is asymptotically stable, then

$$\lim_{T\to\infty} V_1(\underline{s}(T)) = 0.$$

Therefore,

$$\int_0^\infty \dot{V}_1(\underline{s}(t))dt = -\underline{V}_1(\underline{s}(0)).$$

Or, from (3A),

$$\int_0^\infty \underline{s}^T(t)\underline{Q}\underline{s}(t)dt=\underline{s}^T(0)\underline{V}_1\underline{s}(0) \tag{4A}$$

where $\underline{V}_1$ is the solution to the algebraic matrix equation

$$\underline{H}^T\underline{V}_1 + \underline{V}_1\underline{H} = -\underline{Q} \ . \tag{5A}$$

We now consider

$$\frac{d}{dt}[t\underline{s}(t)\underline{V}_1\underline{s}(t)] = \underline{s}^T(t)\underline{V}_1\underline{s}(t) + t[\dot{\underline{s}}^T(t)\underline{V}_1\underline{s}(t) + \underline{s}^T(t)\underline{V}_1\underline{s}(t)]$$

$$= V_1(\underline{s}(t)) + t\dot{V}_1(\underline{s}(t)).$$

Hence

$$\int_0^\infty \frac{d}{dt}[t\underline{s}^T(t)\underline{V}_1\underline{s}(t)] = \int_0^\infty V_1(\underline{s}(t))dt + \int_0^\infty t\dot{V}_1(\underline{s}(t))dt.$$

It can be shown that if $\underline{s}(t)$ is the state vector of an asymptotically stable, free system, then for a finite n,

$$\int_0^\infty d[t^n\underline{s}^T(t)\underline{V}_1\underline{s}(t)] = t^n\underline{s}^T(t)\underline{V}_1\underline{s}(t)\Big|_0^\infty = 0.$$

Therefore

$$0 = \int_0^\infty V_1(\underline{s}(t))dt + \int_0^\infty t\dot{V}_1(\underline{s}(t))dt$$

or

$$\int_0^\infty t\dot{V}_1(\underline{s}(t))dt = -\int_0^\infty V_1(\underline{s}(t))dt$$

or, substituting (3A) for $\dot{V}_1(\underline{s}(t))$,

$$\int_0^\infty t\underline{s}^T(t)\underline{Q}\underline{s}(t)dt = -\int_0^\infty V_1(\underline{s}(t))dt. \tag{6A}$$

Now suppose there exists a scalar $V_2(\underline{s}(t))$ such that

$$V_2(\underline{s}(t)) \triangleq \underline{s}^T(t)\underline{V}_2\underline{s}(t) \tag{7A}$$

and

$$\dot{V}_2(\underline{s}(t)) = V_1(\underline{s}(t)). \tag{8A}$$

Then

$$\dot{V}_2(\underline{s}(t)) = \dot{\underline{s}}^T(t)\underline{V}_2\underline{s}(t) + \underline{s}^T(t)\underline{V}_2\dot{\underline{s}}(t)$$

or

$$\dot{V}_2(\underline{s}(t)) = \underline{s}^T(t)[\underline{H}^T\underline{V}_2 + \underline{V}_2\underline{H}]\underline{s}(t).$$

However, from (8A) and (2A) we have

$$\dot{V}_2(\underline{s}(t)) = \underline{s}^T(t)\underline{V}_1\underline{s}(t).$$

Therefore, we see that

$$\underline{H}^T\underline{V}_2 + \underline{V}_2\underline{H} = \underline{V}_1$$

where, since $\underline{V}_1$ is symmetric, then $\underline{V}_2$ is symmetric. From (8A) we have

$$-\int_0^\infty V_1(\underline{s}(t))dt = -\int_0^\infty \dot{V}_2(\underline{s}(t))dt$$

$$-\int_0^\infty V_1(\underline{s}(t))dt = -\int_0^\infty dV_2(\underline{s}(t))$$

$$-\int_0^\infty V_1(s(t))dt = V_2(\underline{s}(0))$$

$$-\int_0^\infty V_1(\underline{s}(t))dt = \underline{s}^T(0)\underline{V}_2\underline{s}(0).$$

Substituting (6A) yields

$$-\int_0^\infty t\underline{s}^T(t)\underline{Q}\underline{s}(t)dt = \underline{s}^T(0)\underline{V}_2\underline{s}(0) \qquad (9A)$$

where $\underline{V}_2$ is the solution to the algebraic matrix equations

$$\begin{aligned} \underline{H}^T\underline{V}_2 + \underline{V}_2\underline{H} &= \underline{V}_1 \\ \underline{H}^T\underline{V}_1 + \underline{V}_1\underline{H} &= -\underline{Q}. \end{aligned} \qquad (10A)$$

The argument can be continued for higher powers of t. Thus for a stable system (1A) and a symmetric, positive definite $\underline{Q}$,

$$\int_0^\infty t^r\underline{s}^T(t)\underline{Q}\underline{s}(t)dt = (-1)^{r+1}r!\underline{s}^T(0)\underline{V}_{r+1}\underline{s}(0) \qquad (11A)$$

where $\underline{V}_{r+1}$ is the solution to the simultaneous algebraic matrix equations

$$\begin{aligned} \underline{H}^T\underline{V}_{s+1} + \underline{V}_{s+1}\underline{H} &= \underline{V}_s \qquad s = 1,2,\ldots,r \\ &\vdots \\ \underline{H}^T\underline{V}_1 + \underline{V}_1\underline{H} &= -\underline{Q}. \end{aligned} \qquad (12A)$$

APPENDIX B

A METHOD FOR THE DIRECT SOLUTION OF THE SYMMETRIC MATRIX $\underline{V}_{r+1}$

A procedure for generating $\underline{V}_{r+1}$ from the system of simultaneous linear algebraic matrix equations

$$\begin{aligned} \underline{H}^T\underline{V}_{s+1} + \underline{V}_{s+1}\underline{H} &= \underline{V}_s \qquad\qquad s = 1,2,\ldots,r \\ &\vdots \\ \underline{H}^T\underline{V}_1 + \underline{V}_1\underline{H} &= -\underline{Q} \end{aligned}$$

is described in Reference 3. The procedure, which is essentially repeated here with minor amplification and notational changes involves the direct solution for $\underline{V}_{r+1}$ and bypasses the need for solving the (r+1)(n+1)n/2 simultaneous equations.

We first form the matrix $\underline{S}$ as follows:

1) Form the symmetric array of integers

$$\underline{I}_{ss} = \begin{bmatrix} 1 & 2 & 4 & 7 & \cdots & & \\ 2 & 3 & 5 & 8 & & & \\ 4 & 5 & 6 & 9 & & & \\ 7 & 8 & 9 & 10 & & & \\ \vdots & & & & \ddots & & \\ \vdots & & & & & & n(n+1)/2 \end{bmatrix}$$

where n is the dimension of $\underline{H}$, $\underline{Q}$, and $\underline{V}_{r+1}$.

2) Form the n intermediate square matrices, $\underline{S}_i$, i=1,2,...,n whose elements are equal to one and whose order is n(n+1)/2.

3) Proceed along the first row of $\underline{I}_{ss}$ and for each integer, j, that is missing from the sequence $\{1,2,\ldots,n(n+1)/2\}$, set all entries in the j^{th} row and column of $\underline{S}_1$ equal to zero.

4) Proceed row by row through the $\underline{H}$ and $\underline{S}_1$ matrices simultaneously. For the first non-zero entry in the first row of $\underline{S}_1$, enter the value h_{11}, for the second non-zero entry in the first row of $\underline{S}_1$, enter h_{12}, and so forth until all entries of $\underline{H}$ have been used to replace all the non-zero entries of $\underline{S}_1$.

5) Multiply the first column of $\underline{S}_1$ by two.

6) Proceed along the second row of $\underline{I}_{ss}$ as in step three, making the appropriate entries in the $\underline{S}_2$ matrix.

7) Repeat step four for the $\underline{S}_2$ matrix.

8) Repeat step five for the second column of the $\underline{S}_2$ matrix.

9) Continue the procedure until all of the $\underline{S}_i$ matrices have been generated.

10) Form the $\underline{S}$ matrix from

$$\underline{S} = \sum_{i=1}^{n} \underline{S}_i^T \quad .$$

We next form the vector $\underline{v}_{r+1}$ from

$$\underline{v}_{r+1} = \underline{S}^{-(r+1)}\underline{q}$$

where $\underline{v}_{r+1}$ and $\underline{q}$ are the leading diagonal elements of the symmetric $\underline{V}_{r+1}$ and $\underline{Q}$ matrices, respectively. That is,

$$\underline{v}_{r+1} = \begin{bmatrix} v_{11} \\ v_{12} \\ v_{22} \\ v_{13} \\ v_{23} \\ v_{33} \\ \vdots \\ v_{nn} \end{bmatrix} \qquad \underline{q} = \begin{bmatrix} q_{11} \\ q_{12} \\ q_{22} \\ q_{13} \\ q_{23} \\ q_{33} \\ \vdots \\ q_{nn} \end{bmatrix} .$$

The matrix $\underline{V}_{r+1}$ can now be generated from the vector $\underline{v}_{r+1}$ in a simple manner.

APPENDIX C

EVALUATION OF THE QUADRATIC FORM $\underline{s}^T(0)(-1)^{r+1}r!\underline{V}_{r+1}\underline{s}(0)$

Let $\underline{x}$ be an n dimensional, unit length vector. That is

$$\underline{x}^T\underline{x} = 1. \tag{1C}$$

The n dimensional vector $\underline{y}$ whose direction is the same as that of $\underline{x}$ but whose length, ρ, is arbitrary is therefore given by

$$\underline{y} = \rho\underline{x} . \tag{2C}$$

We consider the quadratic form

$$C = \underline{y}^T\underline{Ay} \tag{3C}$$

where $\underline{A}$ is symmetric. Substituting (2C) for $\underline{y}$ yields

$$C = \rho^2\underline{x}^T\underline{Ax} . \tag{4C}$$

Let us constrain $\underline{y}$ to lie on or inside an n dimensional hypersphere of radius R. That is,

$$0 \leq \rho \leq R. \tag{5C}$$

We assume we have no further *a priori* knowledge of the vector $\underline{y}$ and therefore wish to consider its worst case value, that is the value of $\underline{y}$ which causes C to take on its maximum value for a given $\underline{A}$.

We see from (4C) that C is maximized for a given $\underline{A}$ when ρ is a maximum and when the scalar quantity $\underline{x}^T\underline{Ax}$ assumes its maximum value. From (5C) we see that the maximum value for ρ is R. Therefore,

$$C_{max} = R^2 \max\{\underline{x}^T \underline{Ax}\}. \tag{6C}$$

We define the quantity, q as

$$q \triangleq \max\{\underline{x}^T \underline{Ax}\} \tag{7C}$$

It can be shown [8] that the matrix $\underline{A}$ in the form

$$\underline{x}^T \underline{Ax} \tag{8C}$$

where $\underline{x}^T \underline{x} = 1$ and $\underline{A}^T = \underline{A}$, has n real eigenvalues, λ_i and n corresponding real eigenvectors $\underline{x}_i$, $i = 1,2,\ldots,n$. Furthermore, the maximum value of the form (8C) is the largest eigenvalue of $\underline{A}$. That is

$$q = \max\{\underline{x}^T \underline{Ax}\} = \lambda_{max}[\underline{A}]. \tag{9C}$$

Also, the vector which gives this maximum value is the eigenvector corresponding to the largest eigenvalue.

Substituting (9C) into (6C) yields

$$C_{max} = R^2 \lambda_{max}[\underline{A}]. \tag{10C}$$

Thus, C_{max} is the value for C in (3C) which represents the worst case value for $\underline{y}$ for a given $\underline{A}$. We wish to minimize C as a function of $\underline{A}$ for this worst case condition. Since R is a fixed quantity, it plays no role in the minimization of C and we can therefore assume that it is equal to one. This is equivalent to assuming that $\underline{y}$ lies on the unit hypersphere.

We are concerned with minimizing the quantity

$$J = \underline{s}^T(0)(-1)^{r+1} r! \underline{V}_{r+1} \underline{s}(0) \tag{11C}$$

where the matrix $(-1)^{r+1} r! \underline{V}_{r+1}$ is symmetric and positive definite and where $\underline{s}(0)$ represents the worst case initial condition of the state vector $\underline{s}(t)$. We can assume, without loss of generality, that $\underline{s}(0)$ lies on the unit hypersphere. From the argument above, we see that J_1, the maximum value for J for a given parameter set {P}, is given by

$$J_1 = \lambda_{max}[(-1)^{r+1} r! \underline{V}_{r+1}] \tag{12C}$$

and that the worst case initial condition vector, $\underline{s}(0)$, is the eigenvector which corresponds to $\lambda_{max}[(-1)^{r+1} r! \underline{V}_{r+1}]$.

APPENDIX D

OBSERVER THEORY

A. THE NEED FOR OBSERVERS

Given a free plant characterized by the state and output equations

$$\dot{\underline{x}}(t) = \underline{Ax}(t) \tag{1D}$$

$$\underline{y}(t) = \underline{Cx}(t) \tag{2D}$$

where $\underline{x}(t)$ is the nx1 state vector, $\underline{y}(t)$ is the mx1 measurement vector, $\underline{A}$ is a known nxn matrix, and $\underline{C}$ is a known mxn matrix, we wish to know the value of the state $\underline{x}(t)$. If $\underline{C}$ were an nxn matrix representing n linearly independent measurements, then $\underline{C}^{-1}$ would exist and $\underline{x}(t)$ could be found from

$$\underline{x}(t) = \underline{C}^{-1}\underline{y}(t). \tag{3D}$$

This is not usually the case, however, so we will assume that there are fewer linearly independent measurements than there are states. That is, we will assume that m is less than n.

Since $\underline{C}^{-1}$ does not exist for m less than n, an alternative scheme must be found to determine $\underline{x}(t)$. One method of estimating the state vector is to construct a model of the plant and operate it under the same conditions as the plant. Such a model is called an observer. The following sections develop the state equations for two different types of observers.

B. FULL-ORDER OBSERVERS

Let the observer state equations be given by

$$\dot{\underline{z}}(t) = \underline{Az}(t). \tag{4D}$$

The solution to (1D) is given by

$$\underline{x}(t) = \varepsilon^{\underline{A}t}\underline{x}(0) \tag{5D}$$

and the solution to (4D) is given by

$$\underline{z}(t) = \varepsilon^{\underline{A}t}\underline{z}(0). \tag{6D}$$

Therefore, if $\underline{z}(0) = \underline{x}(0)$, the observer state will be identical to the plant state for all t greater than or equal to zero. Since, however, in general $\underline{z}(0)$ does not equal $\underline{x}(0)$, the identical dynamics of (1D) and (4D) would result in the persistence of the initial condition error and the observer state would not provide an accurate estimate of the plant state.

An alternative observer is one which contains not only the plant model but also a feedback forcing function made up from an error signal. Since the only accurate indication we have of the states is the measurement vector $\underline{y}(t)$, let us assume that the error signal represents the difference between the plant measurement vector and an identical "measurement" of the observer state vector. Specifically we assume that the model and the feedback error signal are represented by the observer state equations

$$\dot{\underline{z}}(t) = \underline{Az}(t) + \underline{G}(\underline{y}(t) - \underline{Cz}(t)) \tag{7D}$$

The feedback is provided by the nxm observer gain matrix $\underline{G}$. Rewriting (7D) in terms of $\underline{x}(t)$ and $\underline{z}(t)$ we see that

$$\dot{\underline{z}}(t) = \underline{A}\underline{z}(t) + \underline{GC}(\underline{x}(t) - \underline{z}(t)). \tag{8D}$$

Defining the error state $\underline{e}(t)$ as

$$\underline{e}(t) \triangleq \underline{z}(t) - \underline{x}(t) \tag{9D}$$

and substituting into (8D) yields

$$\dot{\underline{z}}(t) = \underline{A}\underline{z}(t) - \underline{GC}\underline{e}(t). \tag{10D}$$

It is clear, therefore, that by using the proper choice for $\underline{G}$, the error, $\underline{e}(t)$, can be used to drive $\underline{z}(t)$ toward $\underline{x}(t)$.

More precisely, differentiating (9D) with respect to time yields

$$\dot{\underline{e}}(t) = \dot{\underline{z}}(t) - \dot{\underline{x}}(t).$$

Substituting (1D) and (10D) for $\dot{\underline{x}}(t)$ and $\dot{\underline{z}}(t)$, respectively, yields

$$\dot{\underline{e}}(t) = \underline{A}\underline{z}(t) - \underline{GC}\underline{e}(t) - \underline{A}\underline{x}(t)$$

$$\dot{\underline{e}}(t) = \underline{A}(\underline{z}(t) - \underline{x}(t)) - \underline{GC}\underline{e}(t)$$

$$\dot{\underline{e}}(t) = (\underline{A}-\underline{GC})\underline{e}(t). \tag{11D}$$

That is

$$\dot{\underline{e}}(t) = \underline{F}\underline{e}(t) \tag{12D}$$

where we define $\underline{F}$ as

$$\underline{F} \triangleq \underline{A} - \underline{GC}. \tag{13D}$$

The solution to (12D) is given by

$$\underline{e}(t) = \varepsilon^{\underline{F}t}\underline{e}(0). \tag{14D}$$

We see, therefore, that if the eigenvalues of $\underline{F}$ all have negative real parts, then for a finite $\underline{e}(0)$,

$$\lim_{t\to\infty} \underline{e}(t) = 0.$$

It is shown in Ref. 4 that the eigenvalues of the observer can be positioned arbitrarily if the pair $(\underline{C},\underline{A})$ is completely observable. That is if the matrix

$$[\ \underline{C}^T \vdots \underline{A}^T\underline{C}^T \vdots (\underline{A}^T)^2\underline{C}^T \vdots \cdots \vdots (\underline{A}^T)^{n-1}\underline{C}^T\ ]$$

has rank n. From (13D) we see that the eigenvalues of $\underline{F}$ are controlled by the selection of the observer gain matrix $\underline{G}$.

The above argument is easily extended to a plant which is driven by the input $\underline{u}(t)$. The plant state and output equations become

$$\dot{\underline{x}}(t) = \underline{Ax}(t) + \underline{Bu}(t) \qquad (15D)$$

$$\underline{y}(t) = \underline{Cx}(t) \qquad (16D)$$

and the observer state equations become

$$\underline{z}(t) = \underline{Az}(t) + \underline{G}(\underline{y}(t) - \underline{Cz}(t)) + \underline{Bu}(t) \qquad (17D)$$

where $\underline{u}(t)$ is an ℓx1 control vector and $\underline{B}$ is a known nxℓ matrix.

C. REDUCED-ORDER OBSERVERS

The full-order observer possesses a certain amount of redundancy since we estimate some states which are essentially being measured directly. We therefore desire to construct an observer which estimates only those (n-m) states which are not being measured directly. We start with the state and output equations

$$\dot{\underline{x}}(t) = \underline{Ax}(t) + \underline{Bu}(t) \qquad (19D)$$

$$\underline{y}(t) = \underline{Cx}(t). \tag{20D}$$

We define the new state vector $\tilde{\underline{x}}(t)$ so that its first m components are the measurement vector $\underline{y}(t)$ and the remaining (n-m) components are the vector $\underline{w}(t)$. That is

$$\tilde{\underline{x}}(t) \triangleq \begin{bmatrix} \underline{y}(t) \\ \hline \underline{w}(t) \end{bmatrix} . \tag{21D}$$

If $\underline{w}(t)$ is selected so it is some linear transformation of the state $\underline{x}(t)$, namely

$$\underline{w}(t) = \underline{Dx}(t). \tag{22D}$$

Then, in view of (20D) and (22D), (21D) can be written as

$$\underline{x}(t) = \begin{bmatrix} \underline{C} \\ \hline \underline{D} \end{bmatrix} \underline{x}(t). \tag{23D}$$

Furthermore, if $\underline{D}$ is chosen such that

$$\begin{bmatrix} \underline{C} \\ \hline \underline{D} \end{bmatrix}^{-1}$$

exists, where

$$\underline{M} \triangleq \begin{bmatrix} \underline{C} \\ \hline \underline{D} \end{bmatrix}$$

then, the state $\underline{x}(t)$ can be found from $\tilde{\underline{x}}(t)$. Namely

$$\underline{x}(t) = \underline{M}^{-1}\tilde{\underline{x}}(t).$$

Assuming that $\underline{M}^{-1}$ exists, the task becomes one of obtaining an estimate for $\underline{w}(t)$. Writing the state equations for $\underline{x}(t)$ we have

$$\dot{\underline{y}}(t) = \underline{C}\dot{\underline{x}}(t) = \underline{CA}\underline{x}(t) + \underline{CB}\underline{u}(t)$$

$$\dot{\underline{w}}(t) = \underline{D}\dot{\underline{x}}(t) = \underline{DA}\underline{x}(t) + \underline{DB}\underline{u}(t). \tag{26D}$$

That is

$$\dot{\tilde{\underline{x}}}(t) = \underline{MA}\underline{x}(t) + \underline{MB}\underline{u}(t). \tag{27D}$$

Substituting (25D) for $\underline{x}(t)$ yields

$$\dot{\tilde{\underline{x}}}(t) = \underline{MAM}^{-1}\tilde{\underline{x}}(t) + \underline{MB}\underline{u}(t). \tag{28D}$$

Defining

$$\underline{A}' \triangleq \underline{MAM}^{-1}, \quad \underline{B}' = \underline{MB}$$

then $$\dot{\tilde{\underline{x}}} = \underline{A}'\tilde{\underline{x}} + \underline{B}'\underline{u}(t) \tag{29D}$$

which can be written in partitioned matrix form as

$$\begin{bmatrix} \dot{\underline{y}}(t) \\ \hline \dot{\underline{w}}(t) \end{bmatrix} = \left[\begin{array}{c|c} \underline{A}'_{11} & \underline{A}'_{12} \\ \hline \underline{A}'_{21} & \underline{A}'_{22} \end{array}\right] \begin{bmatrix} \underline{y}(t) \\ \hline \underline{w}(t) \end{bmatrix} + \begin{bmatrix} \underline{B}'_1 \\ \hline \underline{B}'_2 \end{bmatrix} \underline{u}(t). \tag{30D}$$

Therefore

$$\dot{\underline{y}}(t) = \underline{A}'_{11}\underline{y}(t) + \underline{A}'_{12}\underline{w}(t) + \underline{B}'_1\underline{u}(t) \tag{31D}$$

and

$$\dot{\underline{w}}(t) = \underline{A}'_{21}\underline{y}(t) + \underline{A}'_{22}\underline{w}(t) + \underline{B}'_2\underline{u}(t). \tag{32D}$$

Since $\underline{y}(t)$ is a measured quantity, its time derivative, $\dot{\underline{y}}(t)$, can be determined. The input $\underline{u}(t)$ can also be considered as a known quantity. Therefore, solving (31D) for $\underline{A}'_{12}\underline{w}(t)$ yields

$$\underline{A}'_{12}\underline{w}(t) = \dot{\underline{y}}(t) - \underline{A}'_{11}\underline{y}(t) - \underline{B}'_1\underline{u}(t)$$

where the elements on the right represent "known" quantities.

The quantity $\underline{A}'_{12}\underline{w}(t)$ can therefore be used as the measurement vector for a full-order observer for the plant whose state equations are given by (32D). We therefore write the observer state equations as

$$\begin{aligned}\dot{\hat{\underline{w}}}(t) = {} & \underline{A}'_{22}\hat{\underline{w}}(t) + \underline{A}'_{21}\underline{y}(t) + \underline{B}'_{2}\underline{u}(t) \\ & + \underline{G}(\dot{\underline{y}}(t) - \underline{A}'_{11}\underline{y}(t) - \underline{B}'_{1}\underline{u}(t) - \underline{A}'_{12}\hat{\underline{w}}(t))\end{aligned} \tag{33D}$$

where $\underline{G}$ is an $(n-m)xm$ matrix, $\hat{\underline{w}}(t)$ is the estimate of $\underline{w}(t)$, and where the first three terms represent the plant model and the last term represents the feedback term. Rewriting (33D) yields

$$\begin{aligned}\dot{\hat{\underline{w}}}(t) = {} & (\underline{A}'_{22} - \underline{GA}'_{12})\hat{\underline{w}}(t) + (\underline{A}'_{21} - \underline{GA}'_{11})\underline{y}(t) \\ & + (\underline{B}'_{2} - \underline{GB}'_{1})\underline{u}(t) + \underline{G}\dot{\underline{y}}(t).\end{aligned} \tag{34D}$$

It has been shown [4] that if $(\underline{C},\underline{A})$ is observable then $(\underline{A}'_{12},\underline{A}'_{22})$ is also observable. Therefore $\underline{G}$, the observer gain matrix, can be selected so that $(\underline{A}'_{22} - \underline{GA}'_{12})$ has arbitrary eigenvalues.

Since differentiation is usually troublesome to perform, it is desirable to eliminate the term $\dot{\underline{y}}(t)$ from (34D). To accomplish this we subtract the term $G\dot{y}(t)$ from both sides of (34D) to get

$$\begin{aligned}\dot{\hat{\underline{w}}}(t) - G\dot{\underline{y}}(t) = {} & (\underline{A}'_{22} - \underline{GA}'_{12})\hat{\underline{w}}(t) + (\underline{A}'_{21} - \underline{GA}'_{11})\underline{y}(t) \\ & + (\underline{B}'_{2} - \underline{GB}'_{1})\underline{u}(t).\end{aligned} \tag{35D}$$

Adding and subtracting $(\underline{A}'_{22} - \underline{GA}'_{12})\underline{Gy}(t)$ from the right side of (35D) yields

$$\dot{\hat{\underline{w}}}(t)-\underline{G}\dot{\underline{y}}(t) = (\underline{A}'_{22}-\underline{G}\underline{A}'_{12})(\hat{\underline{w}}(t)-\underline{G}\underline{y}(t))$$

$$+(\underline{A}'_{21}-\underline{G}\underline{A}'_{11} + (\underline{A}'_{22}-\underline{G}\underline{A}'_{12})\underline{G})\underline{y}(t)$$

$$+(B'_2-GB'_1)\underline{u}(t). \tag{36D}$$

We now define

$$\underline{z}(t) \triangleq \hat{\underline{w}}(t) - \underline{G}\underline{y}(t).$$

Equation (36D) can therefore be rewritten as

$$\dot{\underline{z}}(t) = (\underline{A}'_{22}-\underline{G}\underline{A}'_{12})\underline{z}(t) + (\underline{A}'_{21}-\underline{G}\underline{A}'_{11}+(\underline{A}'_{22}-\underline{G}\underline{A}'_{12})\underline{G})\underline{y}(t)$$

$$+ (\underline{B}'_2-GB'_1)\underline{u}(t) \tag{37D}$$

where $\hat{\underline{w}}(t)$ can be computed from

$$\hat{\underline{w}}(t) = \underline{z}(t) + \underline{G}\underline{y}(t). \tag{38D}$$

APPENDIX E

SAMPLE COMPUTER PROGRAM FOR OBSERVER/CONTROLLER DESIGN

A computer program is given on the following pages which can be used to design an observer/controller in the manner prescribed by the algorithm of Section IID. It is assumed that the plant state and output equations are given by

$$\dot{\underline{x}}(t) = \underline{A}\underline{x}(t) + \underline{B}\underline{u}(t)$$
$$\underline{y}(t) = \underline{C}\underline{x}(t) \tag{1E}$$

and that the observer state and estimate equations are given by

$$\dot{\underline{z}}(t) = (\underline{A}-\underline{G}\underline{C})\underline{z}(t) + \underline{G}\underline{y}(t) + \underline{B}\underline{u}(t). \tag{2E}$$

The control law is given by

$$\hat{\underline{u}}(t) = \underline{K}\underline{z}(t) \tag{3E}$$

and the performance measure to be minimized is given by

$$J = \int_0^\infty t^r[\underline{x}^T(t)\underline{Q}\underline{x}(t) + \underline{e}^T(t)\underline{R}\underline{e}(t)]dt. \tag{4E}$$

The following symbols are explained in the order of their appearance in the input section of the program:

BETA - The initial step length for the minimizing search routine DDIRCT.

EPS - The minimum step length allowed.

D - The minimum value that the sum of the eigenvalues (trace) of the closed-loop system is allowed to take on.

ALPH - The weighting multiplier which is used to assign a cost to any violation of the trace constraint.

N - The order of the matrix $\underline{A}$ in (1E).

JY - The maximum number of searches the minimizing search routine is allowed to perform.

NR - The time multiplier exponent r in the performance measure (4E).

M - The row dimension of $\underline{C}$ in (1E). For full-order observer/controller design M is set to zero.

A - The matrix $\underline{A}$ in (1E).

Q - The matrix $\underline{Q}$ in (4E).

BB - The matrix $\underline{B}$ in (1E).

GG - A vector whose elements are given by the initial estimate of $\underline{G}$ and $\underline{K}$ in the form

$$\begin{bmatrix} \underline{G} \\ \hline \underline{K} \end{bmatrix} .$$

SS - The $\underline{R}$ matrix in (4E).

```
C
C
C     THIS PROGRAM IMPLEMENTS THE DESIGN ALGORITHM OF SECTION IID
C     FOR THE OBSERVER/CONTROLLER.
C
C
      REAL *8GG(8),Q(7,7),A(7,7),ALPH,BETA,EPS,JA,JR(7),JI(7),D,COST,PSI
     1(7,7),VMTRX(7,7),R(7,7),BIGJ,SMLJ,BB(7,7),RR,FBG(7,7),G(7,7),SS(7,
     27),SK
      COMPLEX *16W(7,7),CJR(7)
      COMMON /REAL/ A,Q,ALPH,D,PSI,VMTRX,BB,RR,FBG,G,SS,SK
      COMMON /INT2/ N,NR,K,M,NRT
      EXTERNAL JA
      READ (1,14) BETA,EPS,D,ALPH
      READ (1,15) N,JY,NR,M
      READ (1,14) ((A(I,J),J=1,N),I=1,N)
      READ (1,14) ((Q(I,J),J=1,N),I=1,N)
      READ (1,14) (BB(I,1),I=1,N)
C
      NMMMPN = (N-M)*M+N
      NMM = N-M
      READ (1,14) (GG(I),I=1,NMMMPN)
      READ (1,14) ((SS(I,J),J=1,NMM),I=1,NMM)
      WRITE (6,24)
      WRITE (6,16)
      CALL PRINTM (A,N,N)
      WRITE (6,17)
      CALL PRINTM (Q,N,N)
      WRITE (6,33)
      CALL PRINTM (SS,NMM,NMM)
C
      WRITE (6,18)
      CALL PRINTV (GG,NMMMPN)
      WRITE (6,21) D,ALPH,NR
      K = (2*N-M)*(2*N-M+1)/2
      NR = NR+1
C
C
C        SEARCH FOR THE OPTIMAL SET OF GAINS
C
      CALL DDIRCT (GG,NMMMPN,COST,BETA,.25,EPS,JA,KONVRG,JY,0)
C
C        CHECK THE VALIDITY OF THE SOLUTION
C
      IF (KONVRG.EQ.0) WRITE (6,28)
      IF (KONVRG.EQ.-1) WRITE (6,29)
      IF (KONVRG.GT.0) WRITE (6,30) KONVRG
C
```

```
C         PRINT THE OPTIMAL OBSERVER GAIN MATRIX,THE FEEDBACK GAIN
C         (CONTROLLER) MATRIX, AND THE CORRESPONDING PERFORMANCE
C         MEASURE
C
      NN = N
      NMM = N-M
      N = 2*N-M
      WRITE (6,20)
      WRITE (6,23) COST
      WRITE (6,19)
      CALL PRINTM (G,NMM,M)
      WRITE (6,31)
      CALL PRINTM (FBG,NN,1)
C
C         FIND ALL EIGENVALUES AND EIGENVECTORS OF PMTRX
C
C
      DO 1 I=1,N
C
      DO 1 J=1,N
    1 W(I,J) = DCMPLX(VMTRX(I,J),0.0D0)
C
      CALL DALMAT (W,CJR,N,7,NCAL)
      IF (NCAL.LT.N) GO TO 2
      GO TO 3
    2 WRITE (6,32)
      STOP
C
    3 DO 4 I=1,N
    4 JR(I) = CJR(I)
C
C
      DO 5 I=1,N
C
      DO 5 J=1,N
    5 R(I,J) = W(I,J)
C
C
C         SEARCH FOR THE LARGEST EIGENVALUE OF PMTRX
C
      M = 1
      MB = 1
      BIGJ = JR(1)
    6 M = M+1
      IF (M.GT.N) GO TO 8
      IF (BIGJ.LT.JR(M)) GO TO 7
      GO TO 6
    7 BIGJ = JR(M)
```

```
      MB = M
      GO TO 6
C
C        SEARCH FOR THE SMALLEST EIGENVALUE OF PMTRX
C
    8 M = 1
      MS = 1
      SMLJ = JR(1)
    9 M = M+1
      IF (M.GT.N) GO TO 11
      IF (SMLJ.GT.JR(M)) GO TO 10
      GO TO 9
   10 SMLJ = JR(M)
      MS = M
      GO TO 9
C
C        PRINT THE EIGENVECTOR CORRESPONDING TO THE LARGEST EIGENVALUE
C        OF PMTRX.  THIS REPRESENTS THE WORST CASE INITIAL CONDITIONS.
C
C
   11 WRITE (6,22)
      WRITE (6,23) (R(I,MB),I=1,N)
C
C        PRINT THE EIGENVECTOR CORRESPONDING TO THE SMALLEST EIGENVALUE
C        OF PMTRX.  THIS REPRESENTS THE BEST CASE INITIAL CONDITIONS.
C
C
      WRITE (6,25)
      WRITE (6,23) (R(I,MS),I=1,N)
C
C        FIND THE EIGENVALUES OF THE CLOSED LOOP OBSERVER-CONTROLLER-
C        PLANT
C
C
      DO 12 I=1,N
C
      DO 12 J=1,N
   12 W(I,J) = DCMPLX(PSI(I,J),0.0D0)
C
      CALL DALMAT (W,CJR,N,7,NCAL)
C
      DO 13 I=1,N
      JI(I) = CJR(I)*DCMPLX(0.0D0,-1.0D0)
   13 JR(I) = CJR(I)
C
      WRITE (6,26)
      WRITE (6,27) ((JR(I),JI(I)),I=1,N)
      STOP
```

```
C
   14 FORMAT (8F10.7)
   15 FORMAT (16I5)
   16 FORMAT ('0','THE  A MATRIX')
   17 FORMAT ('0','THE  Q MATRIX')
   18 FORMAT ('0','THE  GG VECTOR')
   19 FORMAT ('0','THE  G MATRIX')
   20 FORMAT ('0',6X,'  JA ')
   21 FORMAT ('0',3X,'D=',E12.3,'       ALPH=',E12.3,'      NR=',I1)
   22 FORMAT ('0',3X,'THE WORST CASE  INITIAL CONDITION VECTOR')
   23 FORMAT (3X,E12.5)
   24 FORMAT ('1')
   25 FORMAT ('0',3X,'THE BEST CASE INITIAL CONDITION VECTOR')
   26 FORMAT ('0',3X,'THE  OBSERVER-CONTROLLER-PLANT EIGENVALUES')
   27 FORMAT (3X,E12.5,3X,E12.5,'J')
   28 FORMAT ('0','THE MINIMIZATION HAS NOT YET CONVERGED')
   29 FORMAT ('0','A PARAMETER ERROR HAS BEEN DETECTED')
   30 FORMAT ('0','THE MINIMIZATION CONVERGED  IN ',I4,' STEPS')
   31 FORMAT ('0','THE K MATRIX')
   32 FORMAT ('0','AN ERROR HAS BEEN DETECTED  IN COMPUTING EVALS/EVECS')
   33 FORMAT ('0','THE  S MATRIX')
C
      END
```

```
      FUNCTION JA (GG)
      REAL *8G(7,7),TERM1(7,7),A(7,7),PSI(7,7),B(28,28),Q(7,7),PMTRX(7,7
     1),VMTRX(7,7),JR(7),BIGJ,SFACT,JA,JP,ALPH,TRF,D,TRV,TERM2(7,7),TERM
     23(7,7),TERM4(7,7),FBG(7,7),FBG2(7,7),GG(8),A22(7,7),A12(7,7),BB(7,
     37),RR,Q111(7,7),Q112(7,7),Q121(7,7),Q122(7,7),Q1(7,7),PSI11(7,7),P
     4SI12(7,7),PSI22(7,7),JI(7),SS(7,7),SK
      COMPLEX *16W(7,7),CJR(7)
      COMMON /REAL/ A,Q,ALPH,D,PSI,VMTRX,BB,RR,FBG,G,SS,SK
      COMMON /INT2/ N,NR,K,M,NRT
      MP1 = M+1
      NP1 = N+1
      NPNMM = 2*N-M
      NMM = N-M
      NPM = N+M
      L = 0
C
      DO 1 I=1,NMM
C
      DO 1 J=1,M
      L = L+1
    1 G(I,J) = GG(L)
C
      J = 0
C
      DO 3 I=1,N
      L = NMM*M+I
      FBG(I,1) = GG(L)
      IF (I.GT.M) GO TO 2
      GO TO 3
    2 J = J+1
      FBG2(J,1) = FBG(I,1)
    3 CONTINUE
C
      CALL MMTR (FBG,TERM4,N,1)
      CALL MULT (BB,TERM4,TERM2,N,1,N)
      CALL MMAD (A,TERM2,PSI11,N,N)
      CALL MMTR (FBG2,TERM3,NMM,1)
      CALL MULT (BB,TERM3,PSI12,N,1,NMM)
      II = 0
C
      DO 4 I=MP1,N
      II = II+1
      JJ = 0
C
      DO 4 J=MP1,N
      JJ = JJ+1
    4 A22(II,JJ) = A(I,J)
C
```

```
      II = 0
C
      DO 5 I=1,M
      II = II+1
      JJ = 0
C
      DO 5 J=MP1,N
      JJ = JJ+1
    5 A12(II,JJ) = A(I,J)
C
      CALL MULT (G,A12,TERM1,NMM,M,NMM)
      CALL MSUB (A22,TERM1,PSI22,NMM,NMM)
C
      DO 6 I=1,N
C
      DO 6 J=1,N
      Q1(I,J) = Q(I,J)
    6 PSI(I,J) = PSI11(I,J)
C
      II = 0
C
      DO 7 I=1,N
      II = II+1
      JJ = 0
C
      DO 7 J=NP1,NPNMM
      JJ = JJ+1
      Q1(I,J) = 0.0D0
    7 PSI(I,J) = PSI12(II,JJ)
C
C
      DO 8 I=NP1,NPNMM
C
      DO 8 J=1,N
      Q1(I,J) = 0.0D0
    8 PSI(I,J) = 0.0D0
C
      II = 0
C
      DO 9 I=NP1,NPNMM
      II = II+1
      JJ = 0
C
      DO 9 J=NP1,NPNMM
      JJ = JJ+1
      Q1(I,J) = SS(II,JJ)
    9 PSI(I,J) = PSI22(II,JJ)
C
```

```
      CALL PDCHK (Q1,NPNMM,IFLAG)
      IF (IFLAG.EQ.1) GO TO 15
      CALL FORMB (PSI,NPNMM,B,K)
      CALL FORMP (B,Q1,NR,NPNMM,PMTRX,IFLAG,K)
      IF (IFLAG.EQ.1) GO TO 15
      SFACT = (-1.0D0)**NR
      CALL SMMULT (SFACT,PMTRX,VMTRX,NPNMM,NPNMM)
      CALL PDCHK (VMTRX,NPNMM,IFLAG)
      IF (IFLAG.EQ.1) GO TO 15
      CALL TTRACE (PSI,NPNMM,TRF)
      JP = 0.0D0
      IF (TRF.GE.D) GO TO 10
      JP = ALPH*(TRF-D)**2
   10 CONTINUE
C
      DO 11 I=1,NPNMM
C
      DO 11 J=1,NPNMM
   11 W(I,J) = DCMPLX(VMTRX(I,J),0.0D0)
C
      CALL DALMAT (W,CJR,NPNMM,7,NCAL)
      IF (NCAL.LT.NPNMM) GO TO 15
C
      DO 12 I=1,NPNMM
      JI(I) = CJR(I)*DCMPLX(0.0D0,-1.0D0)
   12 JR(I) = CJR(I)
C
      MM = 1
      BIGJ = JR(1)
   13 MM = MM+1
      IF (MM.GT.NPNMM) GO TO 14
      IF (BIGJ.LT.JR(MM)) BIGJ=JR(MM)
      GO TO 13
   14 JA = BIGJ+JP
      RETURN
   15 JA = 1.0D05
      RETURN
      END
```

```
      SUBROUTINE FORMB (A,N,B,K)
      DIMENSION A(7,7), B(28,28), BT(28,28), B1(28,28), ISS(7,7)
      DOUBLE PRECISION A,BT,B1,B
      L = 0
      K = (N*(N+1))/2
C
      DO 1 I=1,N
C
      DO 1 J=1,I
      L = L+1
    1 ISS(I,J) = L
C
C
      DO 2 I=1,N
      M = I+1
C
      DO 2 J=M,N
    2 ISS(I,J) = ISS(J,I)
C
C
      DO 3 I=1,K
C
      DO 3 J=1,K
      B1(I,J) = 1.0D0
    3 BT(I,J) = 0.0D0
C
C
      DO 13 M=1,N
      J = 0
C
      DO 5 L=1,K
      J = J+1
      IF (J.GT.N) GO TO 6
      IF (ISS(M,J).EQ.L) GO TO 5
      J = J-1
C
      DO 4 I=1,K
      B1(L,I) = 0.0D0
      B1(I,L) = 0.0D0
    4 CONTINUE
C
    5 CONTINUE
C
      IF (L.EQ.K) GO TO 8
C
    6 DO 7 I=L,K
C
      DO 7 JJ=1,K
```

```
      B1(I,JJ) = 0.0D0
      B1(JJ,I) = 0.0D0
    7 CONTINUE
C
    8 NN = 1
      MM = 1
C
      DO 10 II=1,K
C
      DO 10 JJ=1,K
      IF (B1(II,JJ).NE.1.0D0) GO TO 10
      B1(II,JJ) = A(NN,MM)
      IF (MM.EQ.M) B1(II,JJ)=2.0D0*B1(II,JJ)
      MM = MM+1
      IF (MM.GT.N) GO TO 9
      GO TO 10
    9 NN = NN+1
      MM = 1
      IF (NN.GT.N) GO TO 11
   10 CONTINUE
C
   11 CALL MADD (BT,B1,BT,K,K)
C
      DO 12 KK=1,K
C
      DC 12 LL=1,K
   12 B1(KK,LL) = 1.0D0
C
   13 CONTINUE
C
      CALL MTRA (BT,B,K,K)
      RETURN
      END
```

```
      SUBROUTINE FORMP (B,Q,NR,N,PMTRX,IFLAG,K)
      DIMENSION B(28,28), W(28,28), Q(7,7), QV(28,28), PV(28,28), PMTRX(
     17,7), LI(784), MI(784), WV(784)
      DOUBLE PRECISION Q,PV,QV,PMTRX,W,B,DET,WV
      L = 0
C
      DO 1 J=1,N
C
      DO 1 I=1,J
      L = L+1
    1 QV(L,1) = Q(J,I)
C
      K = N*(N+1)/2
C
      DO 3 I=1,K
C
      DO 3 J=1,K
      IF (I.EQ.J) GO TO 2
      W(I,J) = 0.0D0
      GO TO 3
    2 W(I,J) = 1.0D0
    3 CONTINUE
C
C
      DO 4 I=1,NR
    4 CALL MMULT (W,B,W,K,K,K)
C
      L = 0
C
      DO 5 J=1,K
C
      DO 5 I=1,K
      L = L+1
    5 WV(L) = W(I,J)
C
      CALL PDINV (WV,K,DET,IFLAG,LI,MI)
      IF (IFLAG.EQ.1) GO TO 9
      L = 0
C
      DO 6 J=1,K
C
      DO 6 I=1,K
      L = L+1
    6 W(I,J) = WV(L)
C
      CALL MMULT (W,QV,PV,K,K,1)
      LL = 0
C
```

```
      DO 7 J=1,N
C
      DO 7 I=1,J
      LL = LL+1
    7 PMTRX(J,I) = PV(LL,1)
C
C
      DO 8 I=2,N
      II = I-1
C
      DO 8 J=1,II
    8 PMTRX(J,I) = PMTRX(I,J)
C
    9 RETURN
      END
```

```
      SUBROUTINE PDCHK (PMTRX,N,IFLAG)
      DIMENSION PMV(77), PMTRX(7,7), LI(7), MI(7)
      DOUBLE PRECISION PMV,PMTRX,DET
      L = 0
C
      DO 1 I=1,N
C
      DO 1 J=1,N
      L = L+1
    1 PMV(L) = PMTRX(J,I)
C
      NN = -N
      CALL PDINV (PMV,NN,DET,IFLAG,LI,MI)
      RETURN
      END
```

```
      SUBROUTINE PDINV (A,NN,D,IFLAG,L,M)
C
C
C        SUBROUTINE PDINV
C
C
C        PURPOSE:  TO INVERT OR CHECK FOR POSITIVE DEFINITENESS OF A
C                  SQUARE MATRIX
C
C
C        NOMENCLATURE:
C
C                  A   AN NN**2 VECTOR WHICH HAS BEEN PACKED COLUMN BY
C                  COLUMN FROM AN NN BY NN MATRIX
C
C                  NN   THE ORDER OF THE MATRIX TO BE INVERTED (POSITIVE
C                  INTEGER) OR CHECKED FOR POSITIVE DEFINITENESS
C                  (NEGATIVE INTEGER)
C
C                  D   THE RESULTANT DETERMINANT OF THE MATRIX A
C
C                  IFLAG   AN OUTPUT INDICATOR  FOR POSITIVE DEFINITENESS
C                        (=0 FOR POS. DEF., =1 OTHERWISE) AND FOR
C                        SINGULARITY (=0 FOR NON-SINGULAR, =1
C                  OTHERWISE)
C
C                  L AND M  WORK VECTORS DIMENSIONED IN THE CALLING
C                  PROGRAM TO AT LEAST NN LENGTH
C                           TO AT LEAST NN LENGTH
C
C
      DIMENSION A(1), L(1), M(1)
      DCUBLE PRECISION A,D,BIGA,HOLD
C
C        SEARCH FOR LARGEST ELEMENT
C
      IR = 0
      IF (NN.GT.0) GO TO 1
      NN = -NN
      IR = 1
      LL = 1
      GO TO 2
    1 LL = NN
C
    2 DO 20 N=LL,NN
      IFLAG = 1
      D = 1.0D0
      N = NN
```

```
      NK = -N
C
      DO 19 K=1,N
      NK = NK+N
      L(K) = K
      M(K) = K
      KK = NK+K
      BIGA = A(KK)
C
      DO 4 J=K,N
      IZ = N*(J-1)
C
      DO 4 I=K,N
      IJ = IZ+I
      IF (DABS(BIGA)-DABS(A(IJ))) 3,4,4
    3 BIGA = A(IJ)
      L(K) = I
      M(K) = J
    4 CONTINUE
C
C
C        INTERCHANGE ROWS
C
      J = L(K)
      IF (J-K) 7,7,5
    5 KI = K-N
C
      DO 6 I=1,N
      KI = KI+N
      HOLD = -A(KI)
      JI = KI-K+J
      A(KI) = A(JI)
    6 A(JI) = HOLD
C
C
C        INTERCHANGE COLUMNS
C
    7 I = M(K)
      IF (I-K) 10,10,8
    8 JF = N*(I-1)
C
      DO 9 J=1,N
      JK = NK+J
      JI = JP+J
      HOLD = -A(JK)
      A(JK) = A(JI)
    9 A(JI) = HOLD
C
```

```
C
C        DIVIDE COLUMN BY MINUS PIVOT (VALUE OF PIVOT ELEMENT IS
C        CONTAINED IN BIGA)
C
   10 IF (DABS(BIGA).GT.1.D-30) GO TO 11
      D = 0.0D0
      RETURN
C
   11 DO 13 I=1,N
      IF (I-K) 12,13,12
   12 IK = NK+I
      A(IK) = A(IK)/(-BIGA)
   13 CONTINUE
C
C
C        REDUCE MATRIX
C
C
      DO 16 I=1,N
      IK = NK+I
      HOLD = A(IK)
      IJ = I-N
C
      DO 16 J=1,N
      IJ = IJ+N
      IF (I-K) 14,16,14
   14 IF (J-K) 15,16,15
   15 KJ = IJ-I+K
      A(IJ) = HOLD*A(KJ)+A(IJ)
   16 CONTINUE
C
C
C        DIVIDE ROW BY PIVOT
C
      KJ = K-N
C
      DO 18 J=1,N
      KJ = KJ+N
      IF (J-K) 17,18,17
   17 A(KJ) = A(KJ)/BIGA
C
   18 CONTINUE
C
C        PRODUCT OF PIVOTS
C
      D = D*BIGA
      IF (IR.EQ.1) GO TO 19
C
```

```
C         REPLACE PIVOT BY RECIPROCAL
C
      A(KK) = 1.0D0/BIGA
   19 CONTINUE
C
      IF (IR.EQ.0) GO TO 21
      IF (D.LE.0.0D0) GO TO 29
      IFLAG = 0
   20 CONTINUE
C
C
C         FINAL ROW AND COLUMN INTERCHANGE
C
   21 K = N
      IFLAG = 0
   22 K = (K-1)
      IF (K) 29,29,23
   23 I = L(K)
      IF (I-K) 26,26,24
   24 JQ = N*(K-1)
      JR = N*(I-1)
C
      DO 25 J=1,N
      JK = JQ+J
      HOLD = A(JK)
      JI = JR+J
      A(JK) = -A(JI)
   25 A(JI) = HOLD
C
   26 J = M(K)
      IF (J-K) 22,22,27
   27 KI = K-N
C
      DO 28 I=1,N
      KI = KI+N
      HOLD = A(KI)
      JI = KI-K+J
      A(KI) = -A(JI)
   28 A(JI) = HOLD
C
      GO TO 22
   29 RETURN
      END
```

```
      SUBROUTINE MMULT (A,B,C,L,M,N)
      DIMENSION A(28,28), B(28,28), C(28,28), D(28,28)
      DOUBLE PRECISION A,B,C,SUM,D
C
      DO 2 I=1,L
C
      DO 2 K=1,N
      SUM = 0.0D0
C
      DO 1 J=1,M
    1 SUM = SUM+A(I,J)*B(J,K)
C
    2 D(I,K) = SUM
C
C
      DO 3 I=1,L
C
      DO 3 J=1,N
    3 C(I,J) = D(I,J)
C
      RETURN
      END
```

```
      SUBROUTINE MADD (A,B,C,N,M)
      DIMENSION A(28,28), B(28,28), C(28,28)
      DOUBLE PRECISION A,B,C
C
      DO 1 I=1,N
C
      DO 1 J=1,M
    1 C(I,J) = A(I,J)+B(I,J)
C
      RETURN
      END
```

```
      SUBROUTINE MSUB (A,B,C,N,M)
      DIMENSION A(7,7), B(7,7), C(7,7)
      DOUBLE PRECISION A,B,C
C
      DO 1 I=1,N
C
      DO 1 J=1,M
    1 C(I,J) = A(I,J)-B(I,J)
C
      RETURN
      END
```

```
      SUBROUTINE MTRA (A,B,N,M)
      DIMENSION A(28,28), B(28,28)
      DOUBLE PRECISION A,B
C
      DO 1 I=1,N
C
      DO 1 J=1,N
    1 B(I,J) = A(J,I)
C
      RETURN
      END
```

```
      SUBROUTINE TTRACE (A,N,TRACE)
      DIMENSION A(7,7)
      DOUBLE PRECISION TRACE,A
      TRACE = 0.0D0
C
      DO 1 I=1,N
    1 TRACE = TRACE+A(I,I)
C
      RETURN
      END
```

```
      SUBROUTINE SMMULT (SCALAR,A,B,L,M)
      DIMENSION A(7,7), B(7,7)
      DOUBLE PRECISION A,B,SCALAR
C
      DO 1 I=1,L
C
      DO 1 J=1,M
    1 B(I,J) = SCALAR*A(I,J)
C
      RETURN
      END
```

```
      SUBROUTINE VMULT (A,B,C,L,M)
      DIMENSIUN A(7), B(7), C(7,7)
      DOUBLE PRECISION A,B,C
C
      DO 1 I=1,L
C
      DO 1 J=1,M
    1 C(I,J) = A(I)*B(J)
C
      RETURN
      END
```

```
      SUBROUTINE PRINTM (A,L,M)
      DIMENSION A(7,7)
      DOUBLE PRECISION A
C
      DO 1 I=1,L
      WRITE (6,2) (A(I,J),J=1,M)
    1 CONTINUE
C
      RETURN
C
    2 FORMAT (' ',8(3X,E12.5))
      END
```

```
      SUBROUTINE PRINTV (A,L)
      DIMENSION A(7)
      DOUBLE PRECISION A
C
      DO 1 I=1,L
      WRITE (6,2) A(I)
    1 CONTINUE
C
      RETURN
C
    2 FORMAT (' ',3X,E12.5)
      END
```

```
C        SUBROUTINE ODIRCT
C
C          DOUBLE PRECISION VERSION OF THE SUBROUTINE DIRECT.
C
C        PURPOSE
C          TO LOCATE A LOCAL MINIMUM OF A FUNCTION, S, OF K VARIABLES
C          BY THE METHOD OF DIRECT SEARCH (HOOKE AND JEEVES)
C
C       USAGE
C          CALL DIRECT(PSI,K,SPSI,DELCAP,RHO,DELLC,S,KONVRG,MAXEV,KN)
C
C        DESCRIPTION OF PARAMETERS
C          PSI     - VECTOR OF K INDEPENDENT VARIABLES. IT IS FILLED INI-
C                    TIALLY BY USER WITH FIRST ESTIMATE OF SOLUTION. AT
C                    EXIT FROM DIRECT IT CONTAINS BEST VALUES ATTAINED.
C          K       - NUMBER OF INDEPENDENT VARIABLES OF FUNCTION, S, TO
C                    BE MINIMIZED
C          SPSI    - AT EXIT FROM DIRECT CONTAINS SMALLEST S(PSI) ATTAINED
C          DELCAP  - INITIAL STEP LENGTH (SAME FOR ALL VARIABLES)
C          N.B. DELCAP IS ALTERED BY DIRECT. DO NOT USE NUMERICAL VALUE IN
C          CALLING LIST.
C          RHO     - STEP REDUCTION FACTOR (O.LT.RHO.LT.1). THE VALUE .125
C                    OR .25 IS SUGGESTED
C          DELLC   - END CRITERION. WHEN CURRENT STEP SIZE IS LESS THAN
C                    DELLC, THE SEARCH IS TERMINATED.
C          S       - THE NAME OF AN EXTERNAL FUNCTION, S(PHI), TO BE MINI-
C                    MIZED. A FUNCTION SUBPROGRAM OF THE NAME MUST BE SUP-
C                    PLIED BY THE USER.
C          KONVRG  - AN INDICATOR TO BE TESTED BY USER UPON EXIT FROM
C                    DIRECT. IF KONVERG
C                     =-1, A PARAMETER ERROR WAS DETECTED, FOR EXAMPLE
C                         K.GT.15   OR  K.LE.0,
C                         DELCAP.LE.0,
C                         RHO.LE.0  OR  RHO.GE.1,
C                         DELLC.LE.0. (RETURN IS MADE IMMEDIATELY.)
C                      =0, MAXEV WAS EXCEEDED. (SPSI IS NOT MINIMUM.)
C                      .GT.0, KONVRG = NUMBER OF EVALUATIONS OF S MADE TO
C                         ATTAIN SPSI. (SUCCESS IS INDICATED.)
C          MAXEV   - MAXIMUM NUMBER OF FUNCTION EVALUATIONS USER WILL ALLOW
C                    IN FINDING MINIMUM. IF MAXEV.LE.0, AN EFFECTIVE VALUE
C                    OF 500 WILL BE USED.
C          KN      - AN INDICATOR TO BE USED TO OBTAIN DIAGNOSTIC OUTPUT IF
C                    NECESSARY. IF KN
C                     .LE.-1, OUTPUT OF FUNCTION VALUE AND CORRESPONDING
C                      VARIABLES IS MADE AT THE ORIGIN, AFTER EACH EXPLORE
C                      MOVE, AFTER EACH PATTERN (INCLUDING POST-PATTERN
C                       EXPLORATION)  AND AT EXIT, WITH FINAL VALUE OF
C                       KONVRG.
```

```
C                   =0, NO OUTPUT IS ACCOMPLISHED BY DIRECT.
C                   .GE.1, OUTPUT IS MADE AS IN THE CASE KN.LE.-1, EXCEPT
C                      EXPLORE MOVES ARE UMITTED.
C                  NOTE.. IN CASES KN.NE.0, OUTPUT OF PARAMETERS IS ALSO
C                  MADE BEFORE SEARCH BEGINS. ALL OUTPUT IS ON MEDIUM 6.
C
C       REMARKS
C         MAXIMUM NUMBER OF INDEPENDENT VARIABLES IS 15.
C         NO CONSTRAINTS ARE IMPOSED.
C         IF OUTPUT MEDIUM 6 IS NOT STANDARD, IT MUST BE DESIGNATED ON A
C           CONTROL CARD.
C
C       SUBROUTINES AND FUNCTION SUBPROGRAMS REQUIRED.
C         S(PSI) TO BE DECLARED EXTERNAL IN USERS PROGRAM UNDER ITS
C         PROPER NAME.
C
C
      SUBROUTINE DDIRCT (PSI,K,SPSI,DELCAP,RHO,DELLC,S,KONVRG,MAXEV,KN)
      IMPLICIT REAL*8(A-H,O-Z)
      DIMENSION PSI(K), PHI(28), SLC(28)
      INTEGER EVAL
C
      IF (K.GT.15) GO TO 29
      IF (K) 29,29,1
    1 IF (DELCAP) 29,29,2
    2 IF (RHO) 29,29,3
    3 IF (RHO.GE.1.0D0) GO TO 29
      IF (DELLC) 29,29,4
    4 MAXEVL = MAXEV
      IF (MAXEVL) 5,5,6
    5 MAXEVL = 500
C
C
    6 DO 7 I=1,K
    7 SLC(I) = DELCAP
C
      SPSI = S(PSI)
      EVAL = 1
C
      IF (KN) 8,9,8
    8 WRITE (6,35) DELCAP,RHO,DELLC,MAXEVL,KN,(I,I=1,K)
      WRITE (6,36) SPSI,(PSI(I),I=1,K)
C
    9 SS = SPSI
C
      DO 10 I=1,K
   10 PHI(I) = PSI(I)
C
```

```
      ASSIGN 11 TO IBK
      GO TO 19
C
   11 IF (KN) 12,13,13
   12 WRITE (6,37) SS,(PHI(I),I=1,K)
C
   13 IF (SS.GE.SPSI) GO TO 27
   14 IF (EVAL.GE.MAXEVL) GO TO 30
C
C
      DO 18 I=1,K
      IF (SLC(I)) 15,29,16
   15 IF (PHI(I).GT.PSI(I)) SLC(I)=-SLC(I)
      GO TO 17
   16 IF (PHI(I).LT.PSI(1)) SLC(I)=-SLC(1)
   17 THET = PSI(I)
      PSI(I) = PHI(I)
   18 PHI(I) = 2.0D0*PHI(I)-THET
C
C
      SPSI = SS
      SPHI = S(PHI)
      SS = SPHI
      EVAL = EVAL+1
      ASSIGN 23 TO 1BK
C
C
   19 DO 22 1=1,K
      THET = PHI(I)
      SLCI = SLC(I)
      PHI(I) = THET+SLCI
      SPH1 = S(PHI)
      EVAL = EVAL+1
      IF (SPHI.LT.SS) GO TO 20
      PH1(I) = THET-SLCI
      SPHI = S(PHI)
      EVAL = EVAL+1
      IF (SPH1.GE.SS) GO TO 21
      SLC(I) = -SLC1
   20 SS = SPHI
      GO TO 22
   21 PHI(I) = THET
   22 CONTINUE
C
C
      GO TO IBK, (11,23)
C
   23 IF (KN) 24,25,24
```

```
   24 WRITE (6,38) SS,(PHI(I),I=1,K)
C
   25 IF (SS.GE.SPSI) GO TO 9
C
      DO 26 I=1,K
      IF (DABS(PHI(I)-PSI(I)).GT..5D0*DABS(SLC(I))) GO TO 14
   26 CONTINUE
C
C
   27 IF (DELCAP.LT.DELLC) GO TO 31
      DELCAP = RHO*DELCAP
C
      DO 28 I=1,K
   28 SLC(I) = RHO*SLC(I)
C
      GO TO 9
C
   29 KONVRG = -1
      GO TO 32
   30 KONVRG = 0
      GO TO 32
   31 KONVRG = EVAL
   32 IF (KN) 33,34,33
   33 WRITE (6,39) KONVRG,SPSI,(PSI(I),I=1,K)
   34 RETURN
C
   35 FORMAT (14H1DIRECT SEARCH,2X,8HDELCAP =,E15.6,2X,5HRHO =,E15.6,2X,
     17HDELLC =,E15.6,2X,8HMAXEVL =,I8,2X,5H KN =,I3//8H0 MOVE   ,15H FUN
     2CTION VALUE,3X,3X,I2,6HST VAR,4X,3X,I2,6HND VAR,4X,3X,I2,6HRD VAR,
     34X,3(3X,I2,6HTH VAR,4X)/26X,6(3X,I2,6HTH VAR,4X)/26X,6(3X,I2,6HTH
     4VAR,4X))
   36 FORMAT (8H0ORIGIN ,E15.7,3X,6E15.6/(26X,6E15.6))
   37 FORMAT (8H0EXPLORE,E15.7,3X,6E15.6/(26X,6E15.6))
   38 FORMAT (8H PATTERN,E15.7,3X,6E15.6/(26X,6E15.6))
   39 FORMAT (10H0KONVRG = ,I10/8H EXIT    ,E15.7,3X,6E15.6/(26X,6E15.6))
      END
```

```
      SUBROUTINE MMTR (A,B,N,M)
      DIMENSION A(7,7), B(7,7)
      DOUBLE PRECISION A,B
C
      DO 1 I=1,N
C
      DO 1 J=1,N
    1 B(I,J) = A(J,I)
C
      RETURN
      END
```

```
      SUBROUTINE MMAD (A,B,C,N,M)
      DIMENSION A(7,7), B(7,7), C(7,7)
      DOUBLE PRECISION A,B,C
C
      DO 1 I=1,N
C
      DO 1 J=1,M
    1 C(I,J) = A(I,J)+B(I,J)
C
      RETURN
      END
```

```
      SUBROUTINE MULT (A,B,C,L,M,N)
      DIMENSION A(7,7), B(7,7), C(7,7), D(7,7)
      DOUBLE PRECISION A,B,C,SUM,D
C
      DO 2 I=1,L
C
      DO 2 K=1,N
      SUM = 0.0D0
C
      DO 1 J=1,M
    1 SUM = SUM+A(I,J)*B(J,K)
C
    2 D(I,K) = SUM
C
C
      DO 3 I=1,L
C
      DO 3 J=1,N
    3 C(I,J) = D(I,J)
C
      RETURN
      END
```

LIST OF REFERENCES

1. Luenberger, D. G., "Observing the State of a Linear System," IEEE Trans. on Military Electronics, Vol. MIL-8, p. 74-80, April 1964.

2. Luenberger, D. G., "An Introduction to Observers," IEEE Trans. on Automatic Control, Vol. AC-16, p. 596-602, December 1971.

3. MacFarlane, A. G. J., "The Calculation of Functionals of the Time and Frequency Response of a Linear Constant Coefficient Dynamical System," Quarterly Journal of Mechanics and Applied Mathematics, Vol. XVI, p. 259-271, 1963.

4. Kalman, R. E., Bertram, J. E., "Control System Analysis and Design Via the 'Second Method' of Lyapunov - 1. Continuous Time Systems," Journal of Basic Engineering, Transactions of the ASME, p. 371-393, June 1960.

5. Levine, W. S., Athans, M., "On the Determination of the Optimal Constant Output Feedback Gains for Linear Multivariable Systems," IEEE Trans. on Automatic Control, Vol. AC-15, p. 44-48, February 1970.

6. Park, C., "Time Domain Design of Observers," MSEE Thesis, Naval Postgraduate School, Monterey, California, 1973.

7. Kreindler, E., Jameson, A., "Conditions for Nonnegativeness of Partitioned Matrices," IEEE Trans. of Automatic Control, Vol. AC-17, p. 146, February 1972.

8. Bodewig, E., Matrix Calculus, Interscience Publishers, Inc. 1959.

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	2
2.	Library, Code 0212 Naval Postgraduate School Monterey, California 93940	2
3.	Department Chairman, Code 52 Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	2
4.	Assoc Professor D. E. Kirk, Code 52Ki Department of Electrical Engineering Naval Postgraduate School Monterey, California 93940	3
5.	LT Robert Ernest LaRock, USN SUPSHIP Newport News Shipbuilding and Drydock Co. Newport News, Virginia 23607	1

Thesis 156347
L269 LaRock
c.1 Optimal design of
observer/controllers.